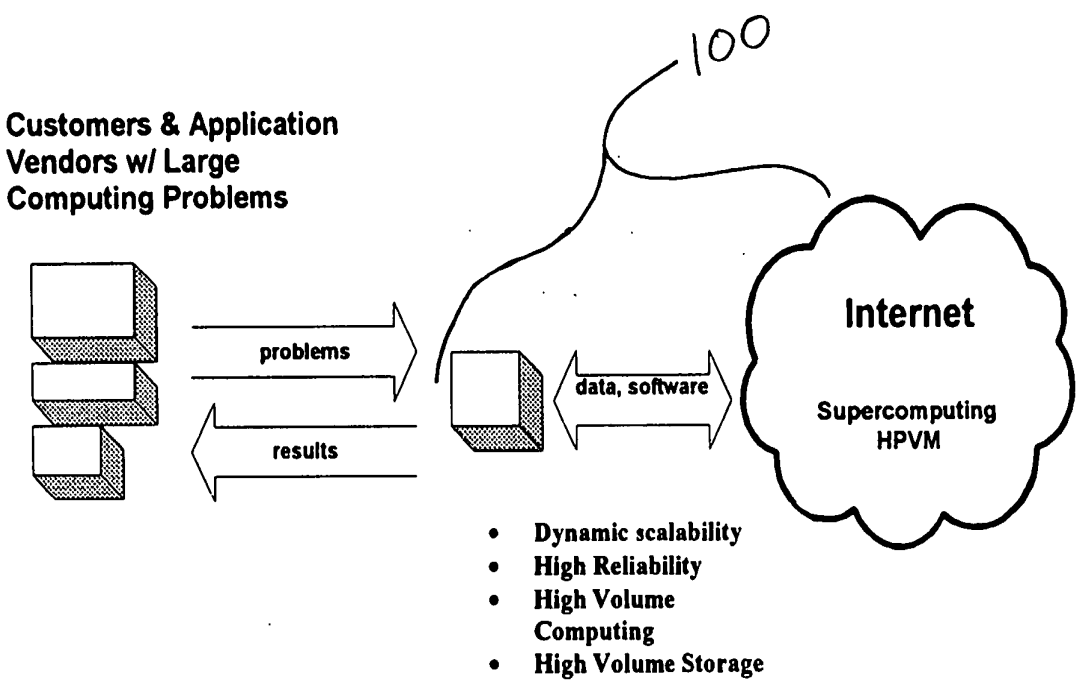
100
Customers & Application Vendors w/ Large Computing Problems
problems
results
data, software
Internet
Supercomputing HPVM
• Dynamic scalability
• High Reliability
• High Volume Computing
• High Volume Storage

FIG. 1

TOZTTO"SE965260

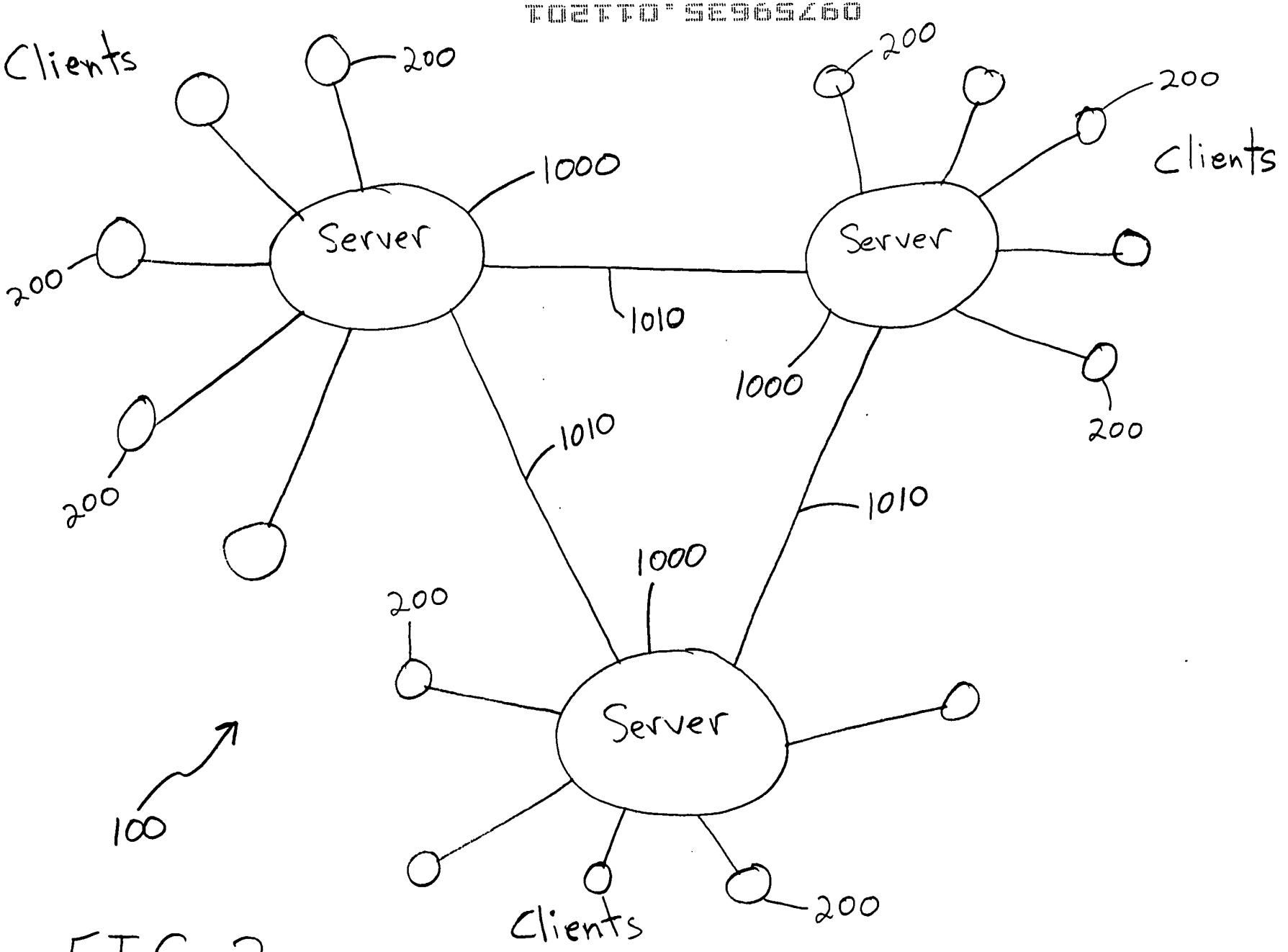

FIG. 2

D2FF0 SE965260

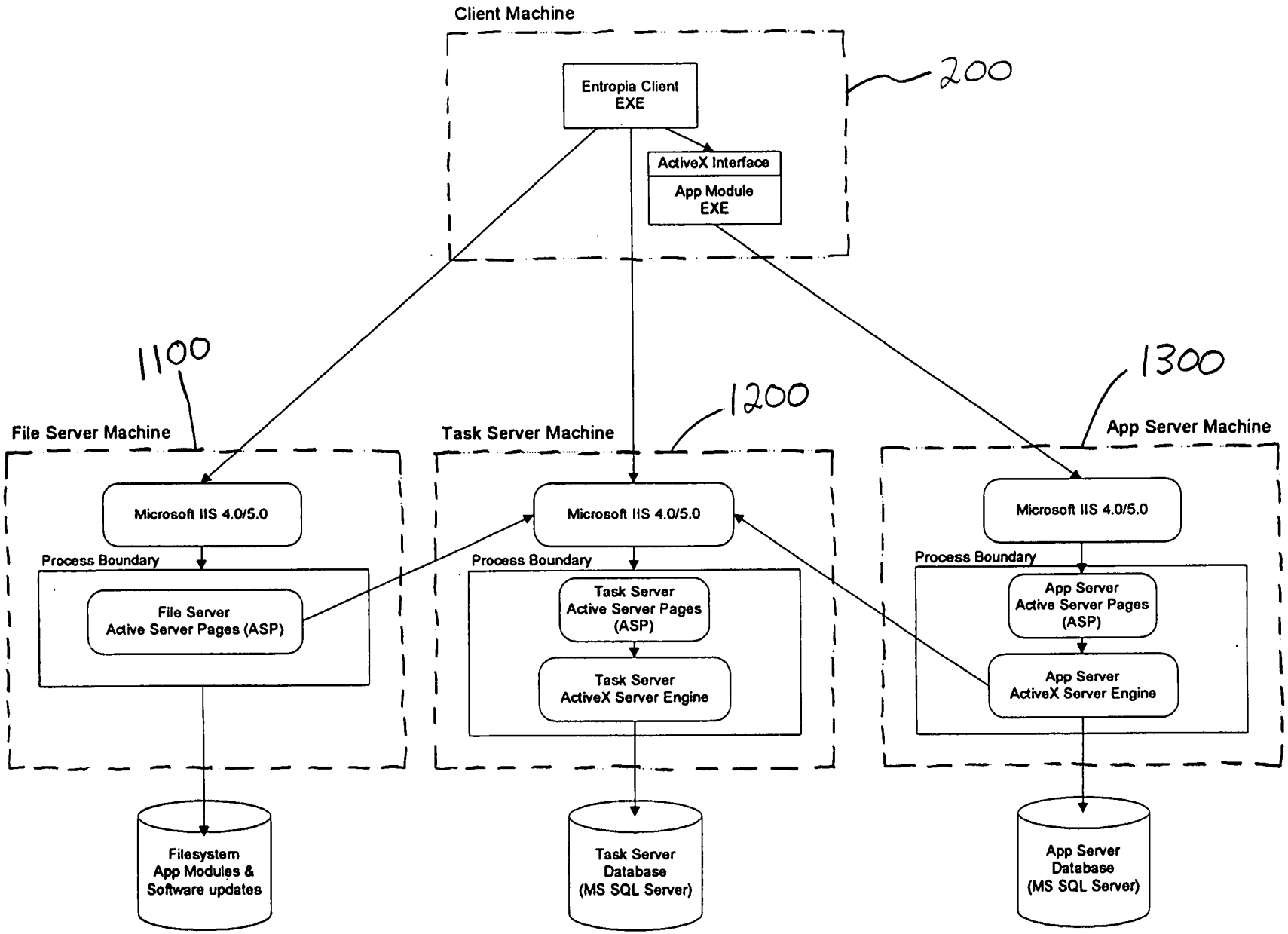

FIG. 3

TOZITO "SE565260

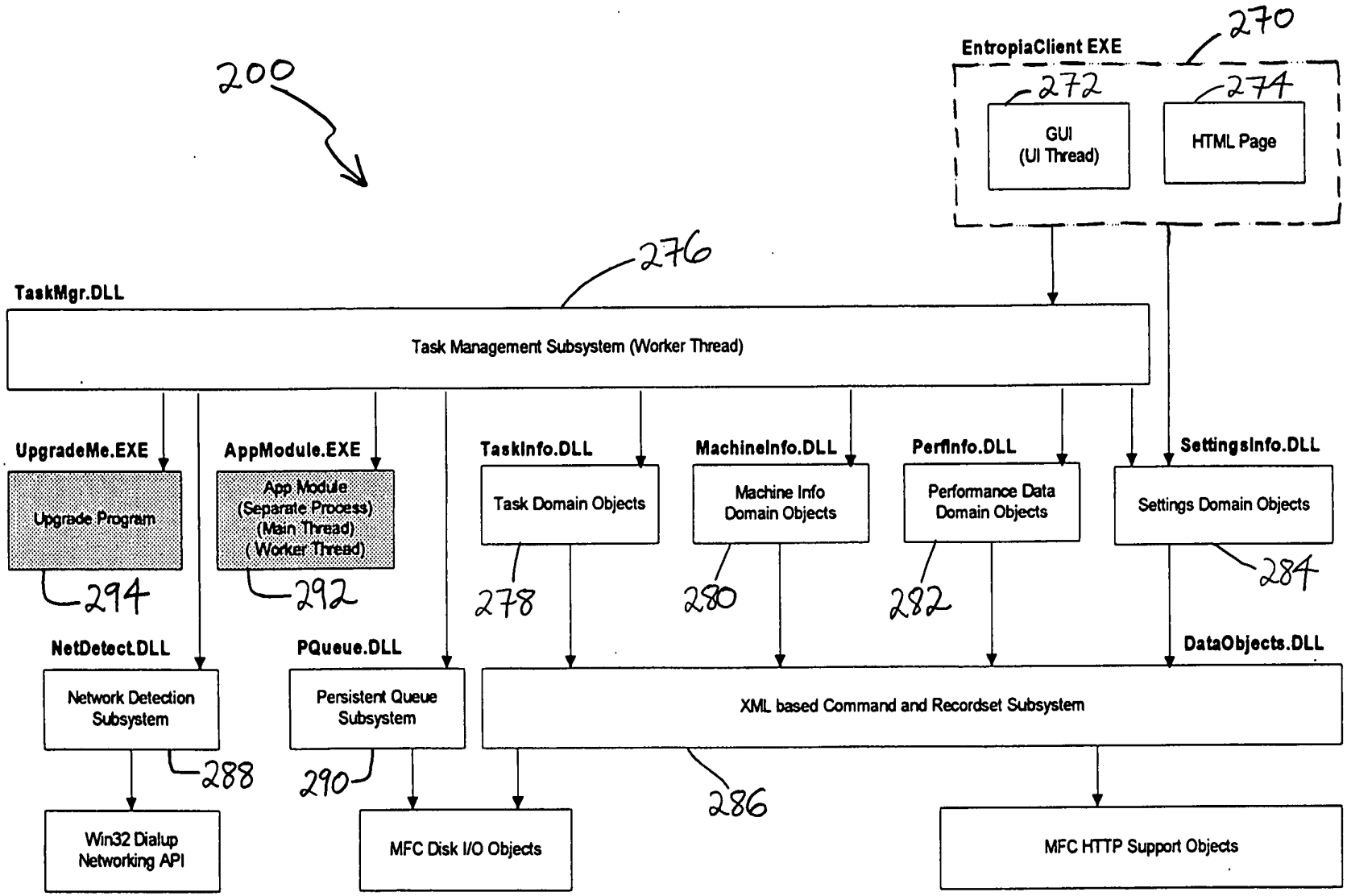

FIG. 4

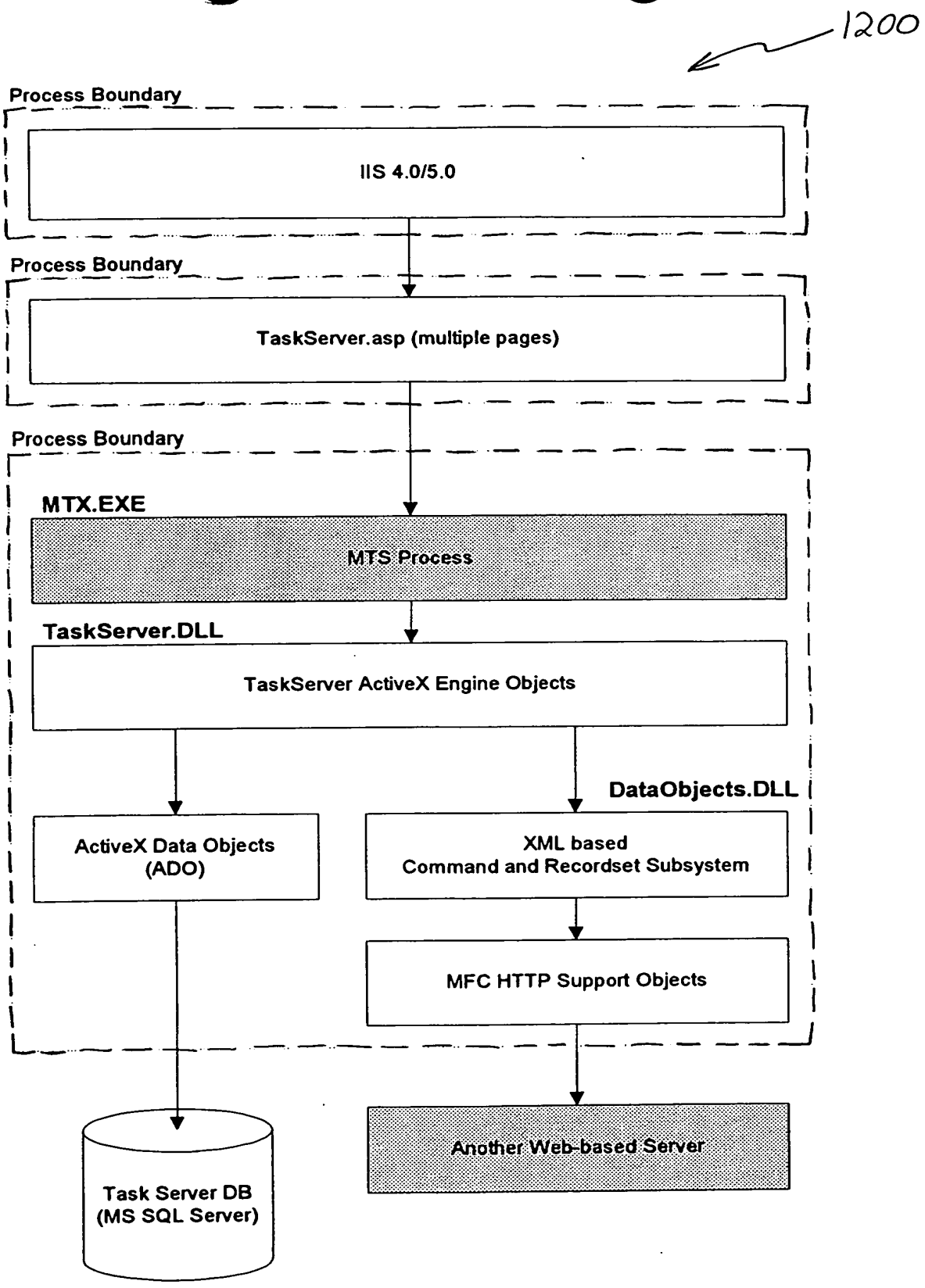
1200
Process Boundary
IIS 4.0/5.0
Process Boundary
TaskServer.asp (multiple pages)
Process Boundary
MTX.EXE
MTS Process
TaskServer.DLL
TaskServer ActiveX Engine Objects
DataObjects.DLL
ActiveX Data Objects (ADO)
XML based Command and Recordset Subsystem
MFC HTTP Support Objects
Another Web-based Server
Task Server DB (MS SQL Server)

FIG. 5

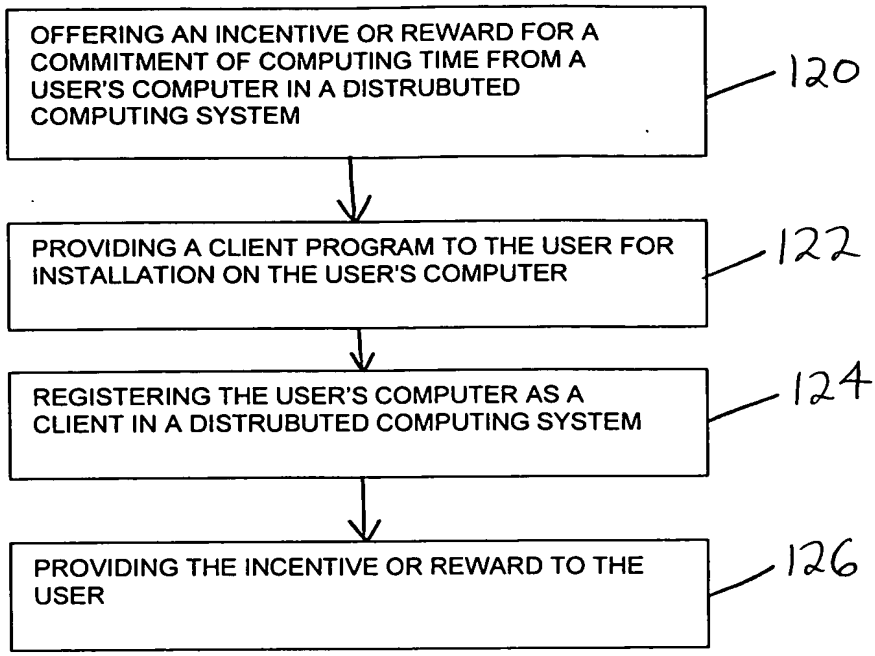
OFFERING AN INCENTIVE OR REWARD FOR A COMMITMENT OF COMPUTING TIME FROM A USER'S COMPUTER IN A DISTRUBUTED COMPUTING SYSTEM
120
PROVIDING A CLIENT PROGRAM TO THE USER FOR INSTALLATION ON THE USER'S COMPUTER
122
REGISTERING THE USER'S COMPUTER AS A CLIENT IN A DISTRUBUTED COMPUTING SYSTEM
124
PROVIDING THE INCENTIVE OR REWARD TO THE USER
126

FIG. 6A

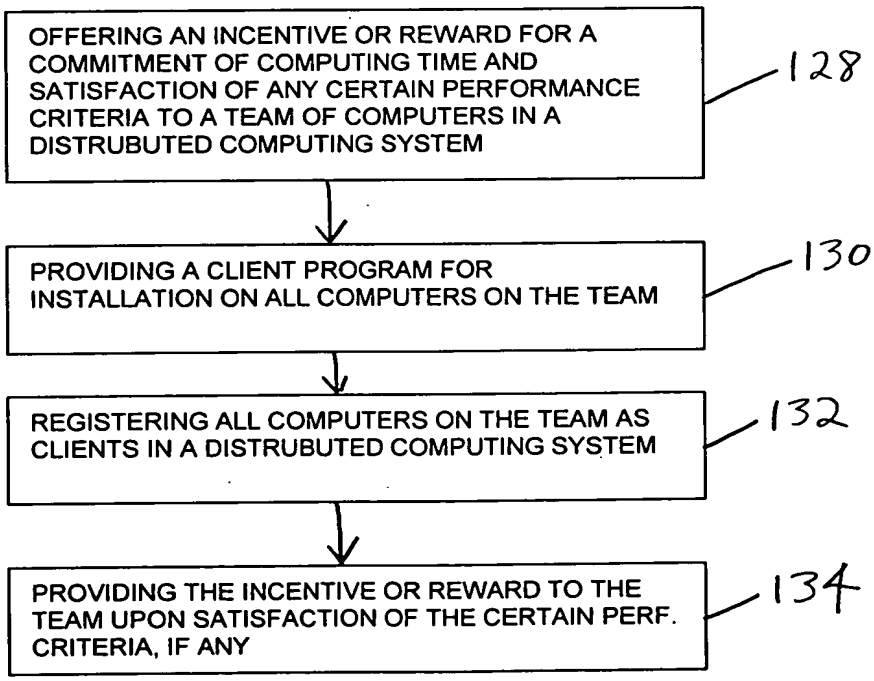
OFFERING AN INCENTIVE OR REWARD FOR A COMMITMENT OF COMPUTING TIME AND SATISFACTION OF ANY CERTAIN PERFORMANCE CRITERIA TO A TEAM OF COMPUTERS IN A DISTRUBUTED COMPUTING SYSTEM
128
PROVIDING A CLIENT PROGRAM FOR INSTALLATION ON ALL COMPUTERS ON THE TEAM
130
REGISTERING ALL COMPUTERS ON THE TEAM AS CLIENTS IN A DISTRUBUTED COMPUTING SYSTEM
132
PROVIDING THE INCENTIVE OR REWARD TO THE TEAM UPON SATISFACTION OF THE CERTAIN PERF. CRITERIA, IF ANY
134

FIG. 6B

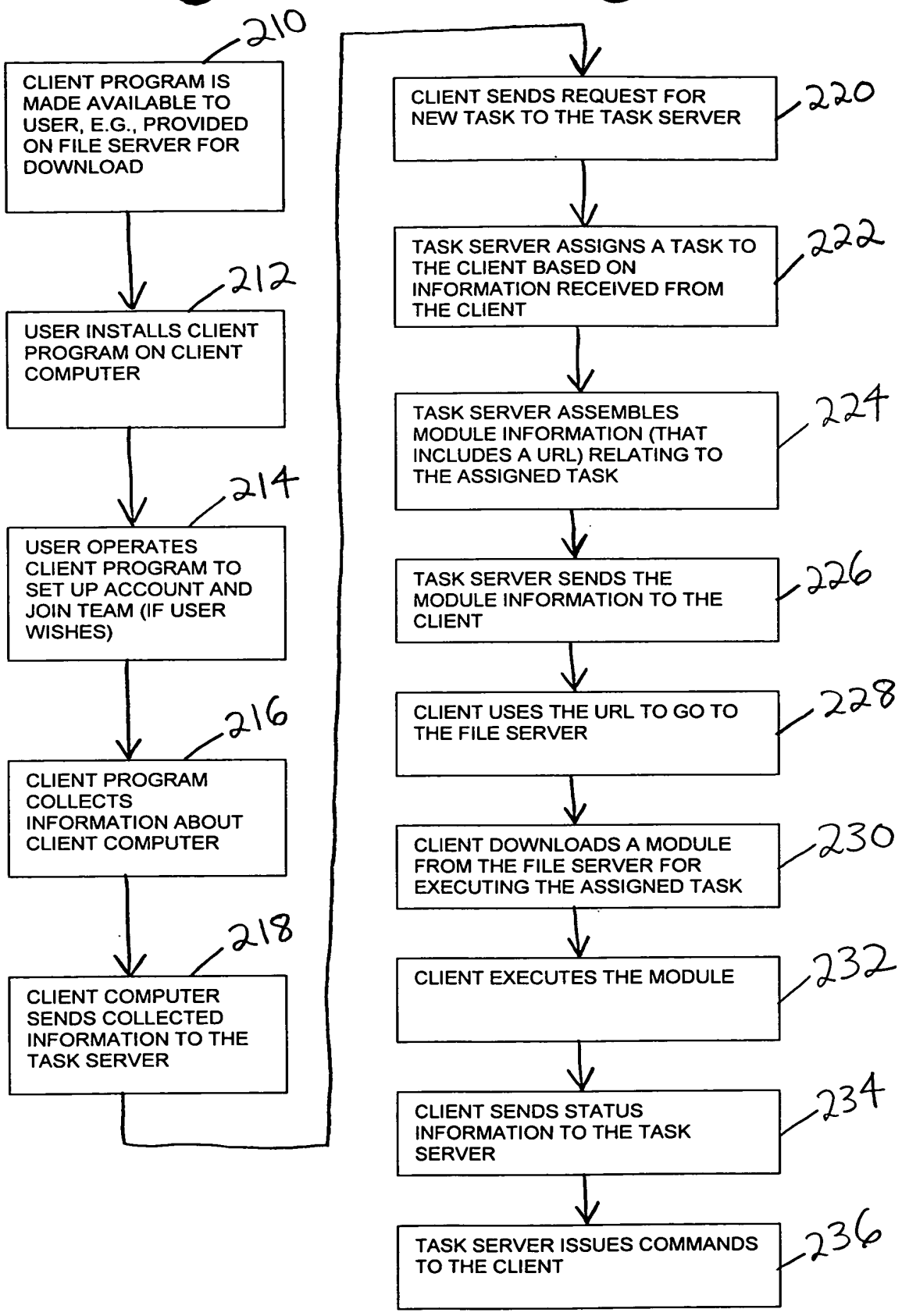
210
CLIENT PROGRAM IS MADE AVAILABLE TO USER, E.G., PROVIDED ON FILE SERVER FOR DOWNLOAD
212
USER INSTALLS CLIENT PROGRAM ON CLIENT COMPUTER
214
USER OPERATES CLIENT PROGRAM TO SET UP ACCOUNT AND JOIN TEAM (IF USER WISHES)
216
CLIENT PROGRAM COLLECTS INFORMATION ABOUT CLIENT COMPUTER
218
CLIENT COMPUTER SENDS COLLECTED INFORMATION TO THE TASK SERVER
220
CLIENT SENDS REQUEST FOR NEW TASK TO THE TASK SERVER
222
TASK SERVER ASSIGNS A TASK TO THE CLIENT BASED ON INFORMATION RECEIVED FROM THE CLIENT
224
TASK SERVER ASSEMBLES MODULE INFORMATION (THAT INCLUDES A URL) RELATING TO THE ASSIGNED TASK
226
TASK SERVER SENDS THE MODULE INFORMATION TO THE CLIENT
228
CLIENT USES THE URL TO GO TO THE FILE SERVER
230
CLIENT DOWNLOADS A MODULE FROM THE FILE SERVER FOR EXECUTING THE ASSIGNED TASK
232
CLIENT EXECUTES THE MODULE
234
CLIENT SENDS STATUS INFORMATION TO THE TASK SERVER
236
TASK SERVER ISSUES COMMANDS TO THE CLIENT

FIG. 7

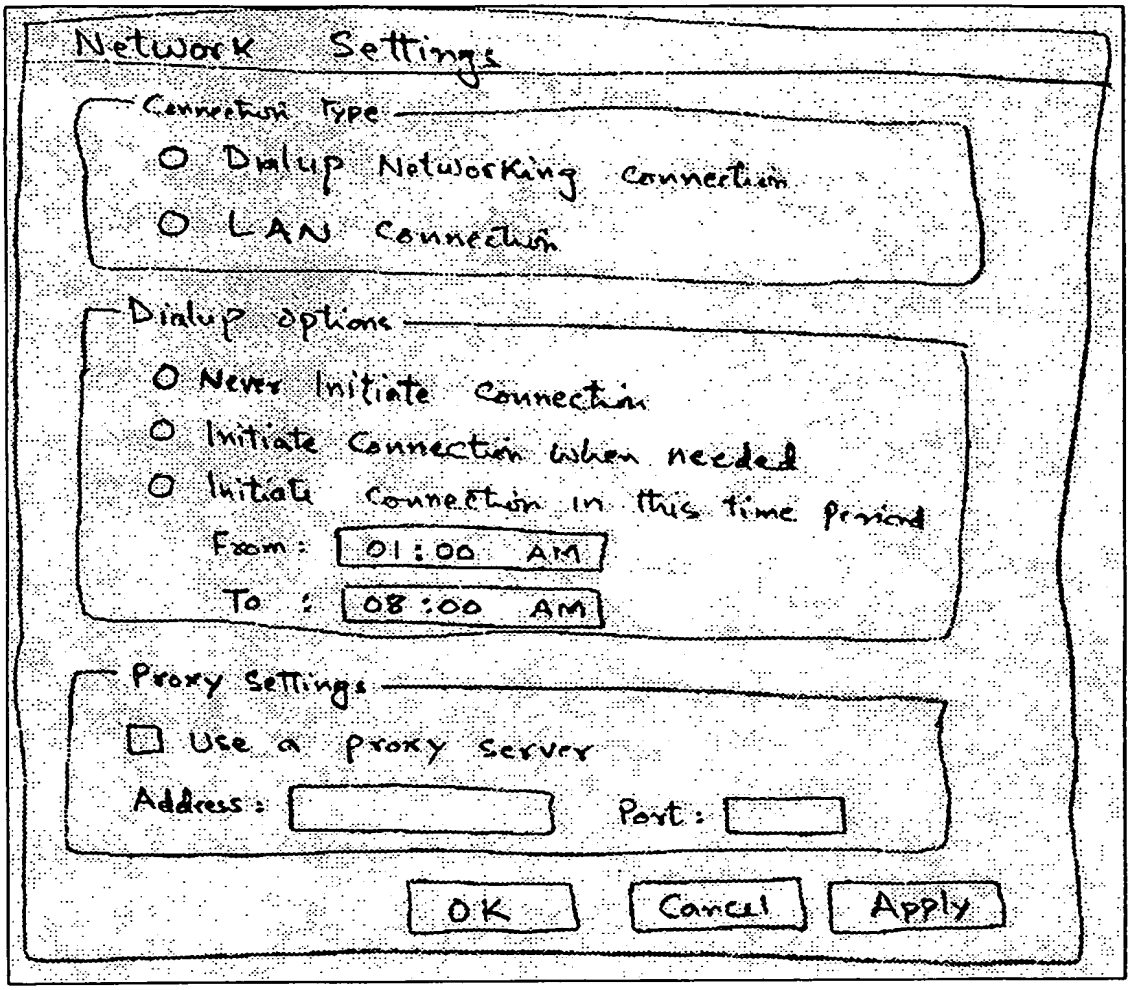
Network Settings
Connection Type
Dialup Networking Connection
LAN Connection
Dialup options
Never Initiate Connection
Initiate Connection when needed
Initiate Connection in this time period
From: 01:00 AM
To: 08:00 AM
Proxy Settings
Use a proxy server
Address:
Port:
OK
Cancel
Apply

FIG. 8

TOZETTO 2250560

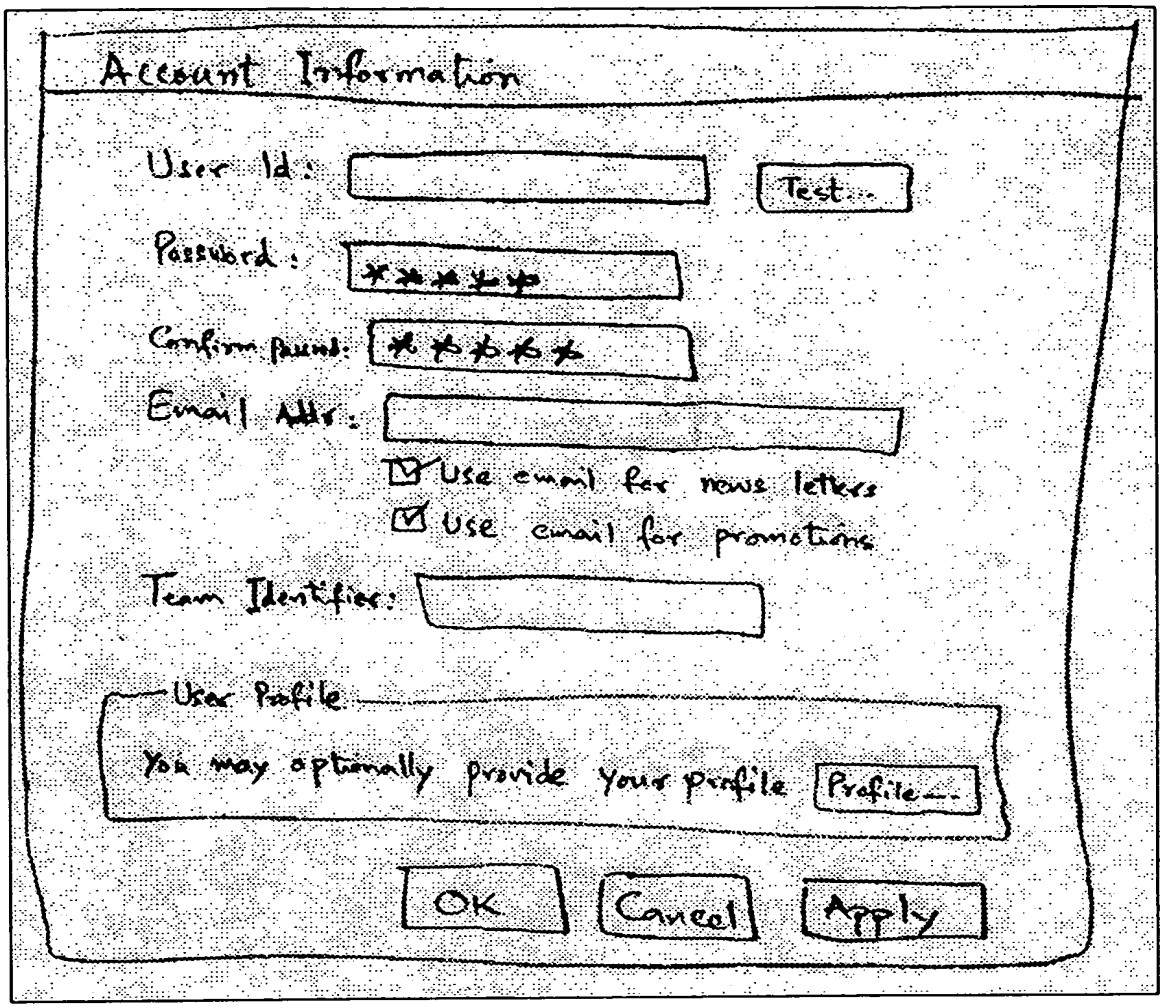

FIG. 9

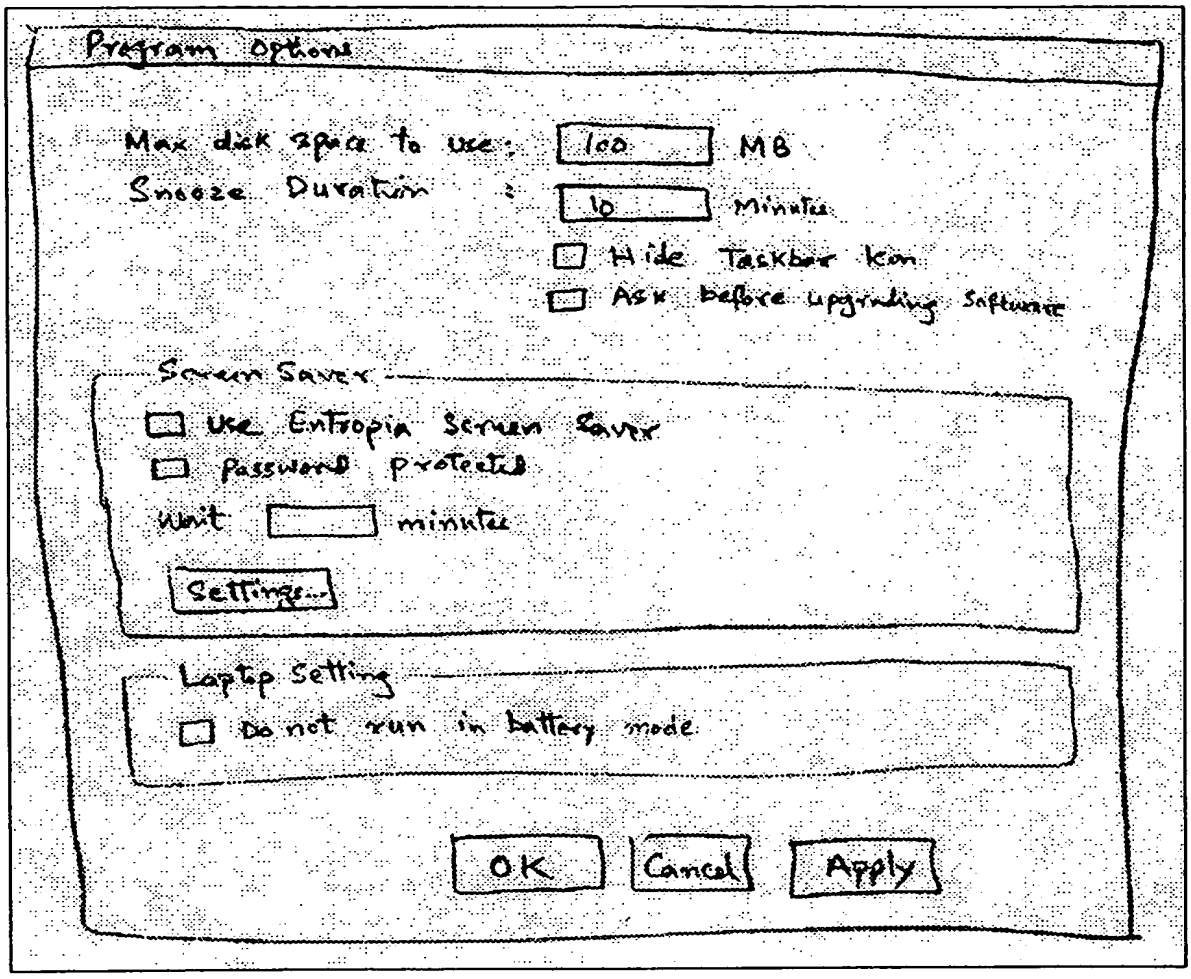
Program Options
Max disk space to use: 100 MB
Snooze Duration : 10 minutes
Hide Taskbar Icon
Ask before upgrading Software
Screen Saver
Use Entropia Screen Saver
Password protected
Wait minutes
Settings...
Laptop Setting
Do not run in battery mode
OK
Cancel
Apply

FIG. 10

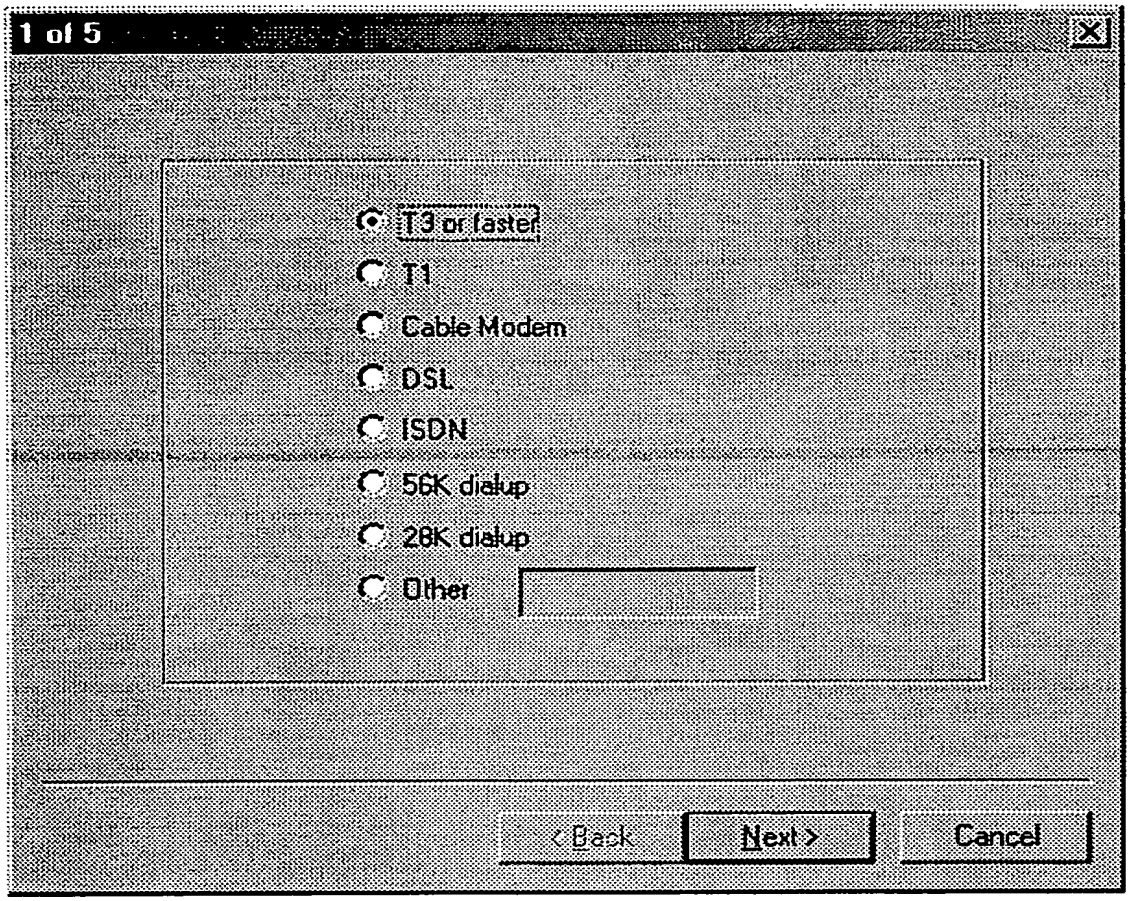
1 of 5
T3 or faster
T1
Cable Modem
DSL
ISDN
56K dialup
28K dialup
Other
< Back
Next >
Cancel

FIG. 11

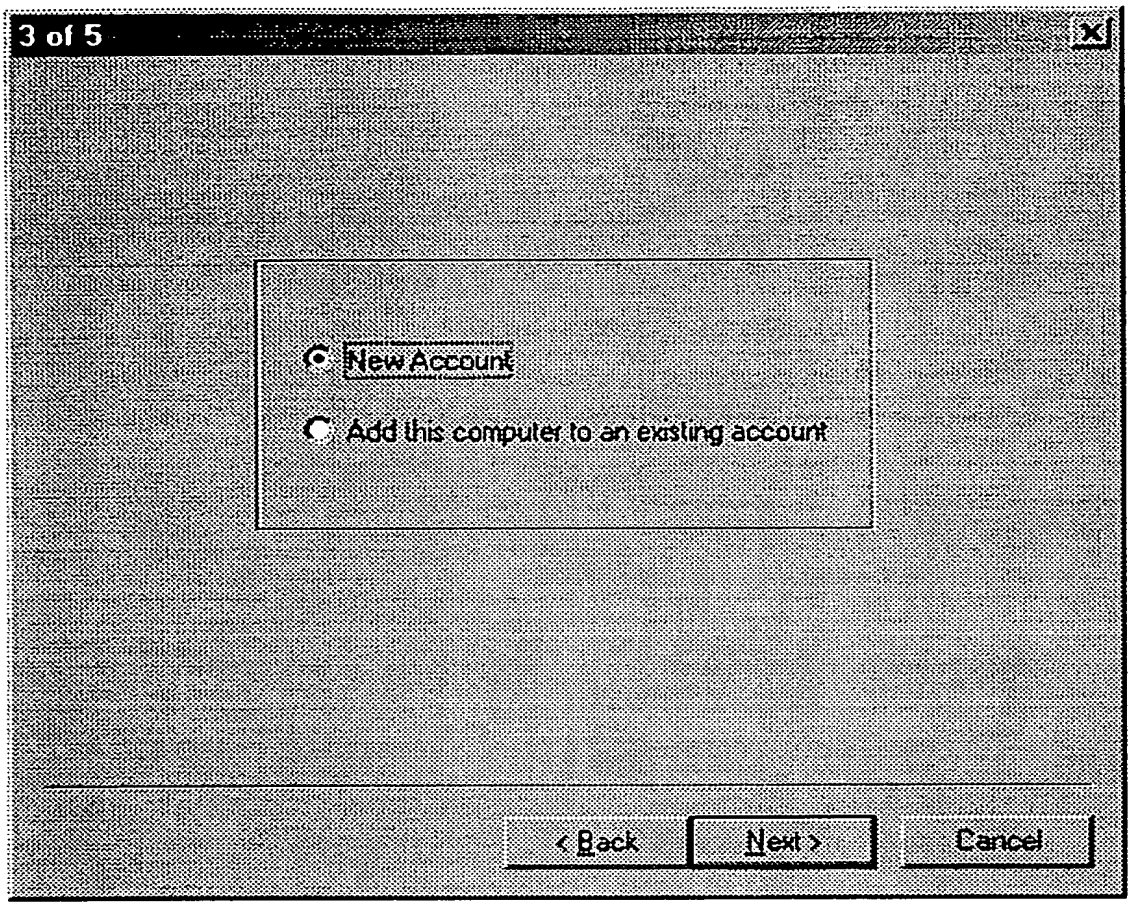
3 of 5
New Account
Add this computer to an existing account
< Back
Next >
Cancel

FIG. 12

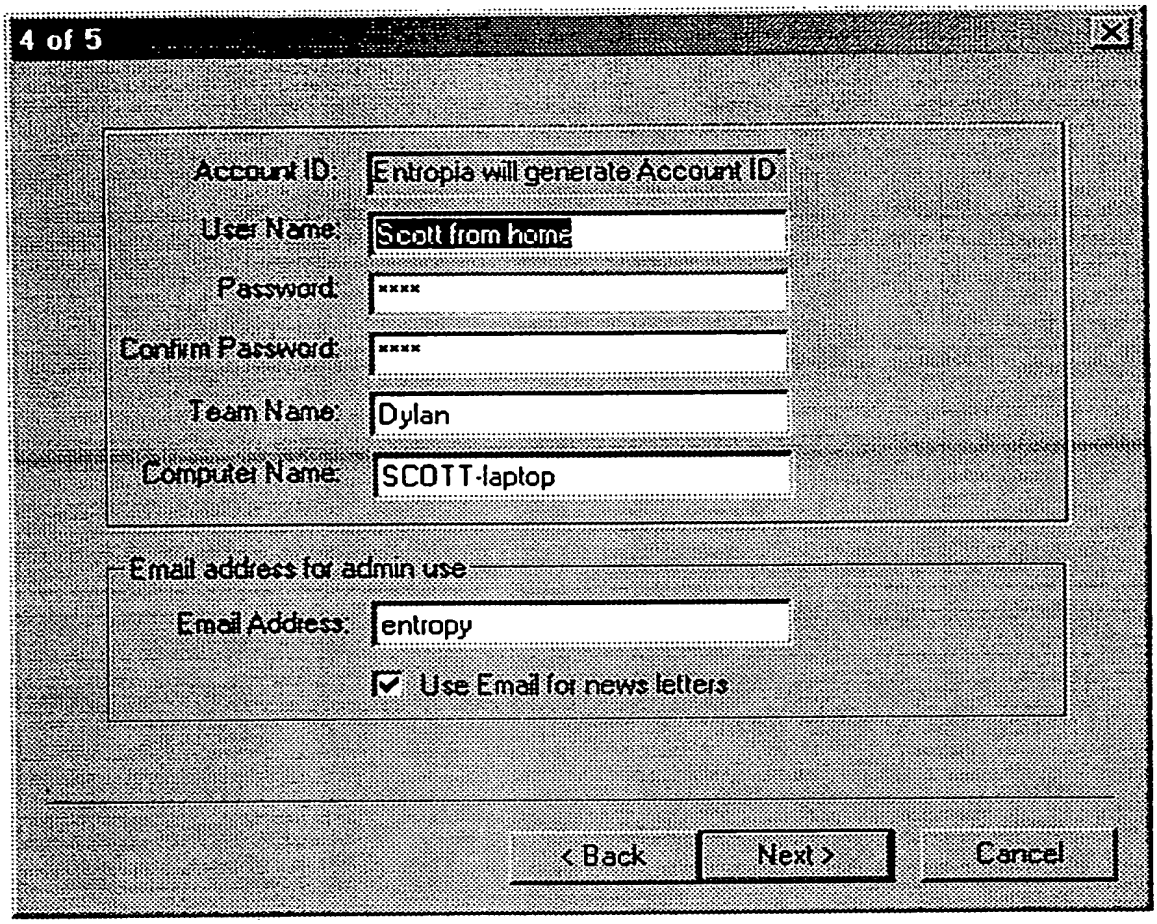
4 of 5
Account ID:
Entropia will generate Account ID
User Name:
Scott from home
Password:
****
Confirm Password:
****
Team Name:
Dylan
Computer Name:
SCOTT-laptop
Email address for admin use
Email Address:
entropy
Use Email for news letters
< Back
Next >
Cancel

FIG. 13

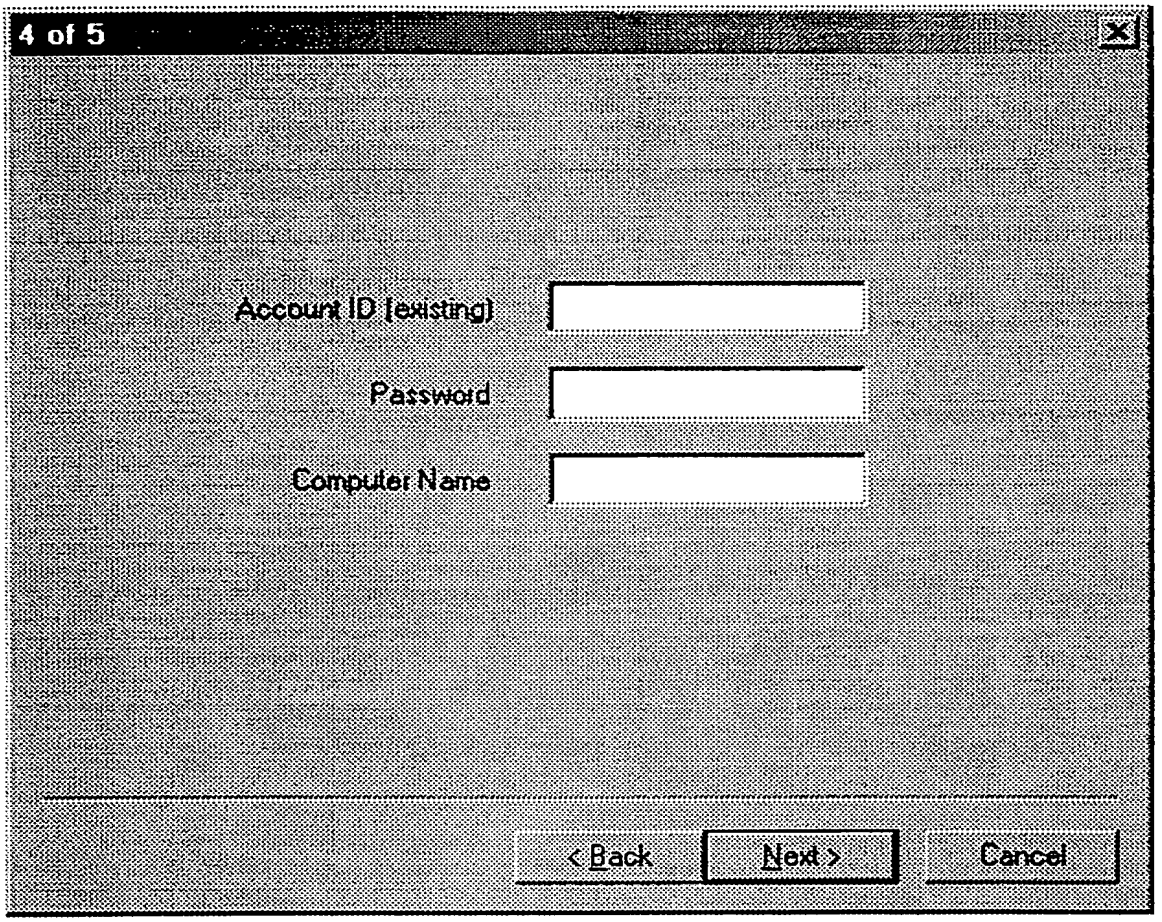
4 of 5
Account ID (existing)
Password
Computer Name
< Back
Next >
Cancel

FIG. 14

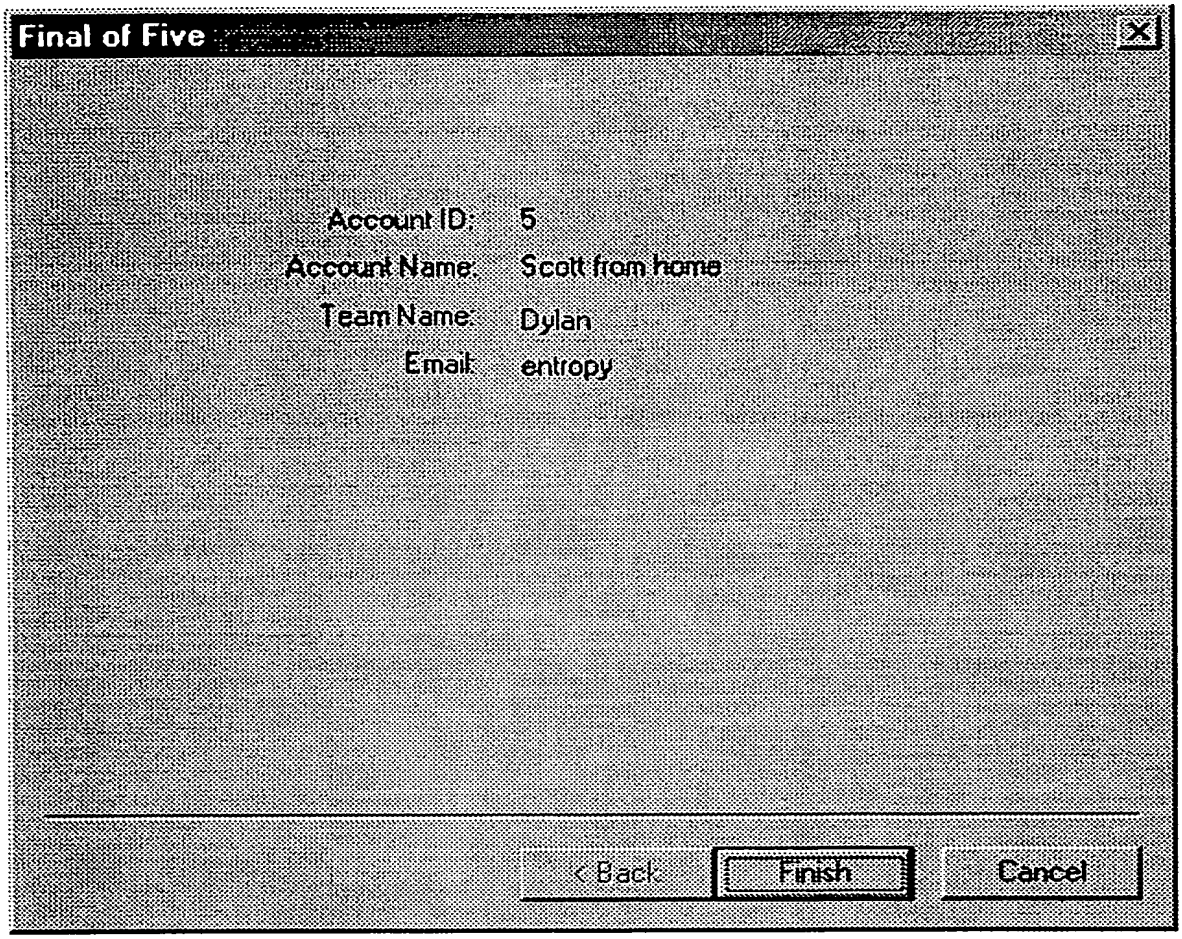
Final of Five
Account ID: 5
Account Name: Scott from home
Team Name: Dylan
Email: entropy
< Back
Finish
Cancel

FIG. 15

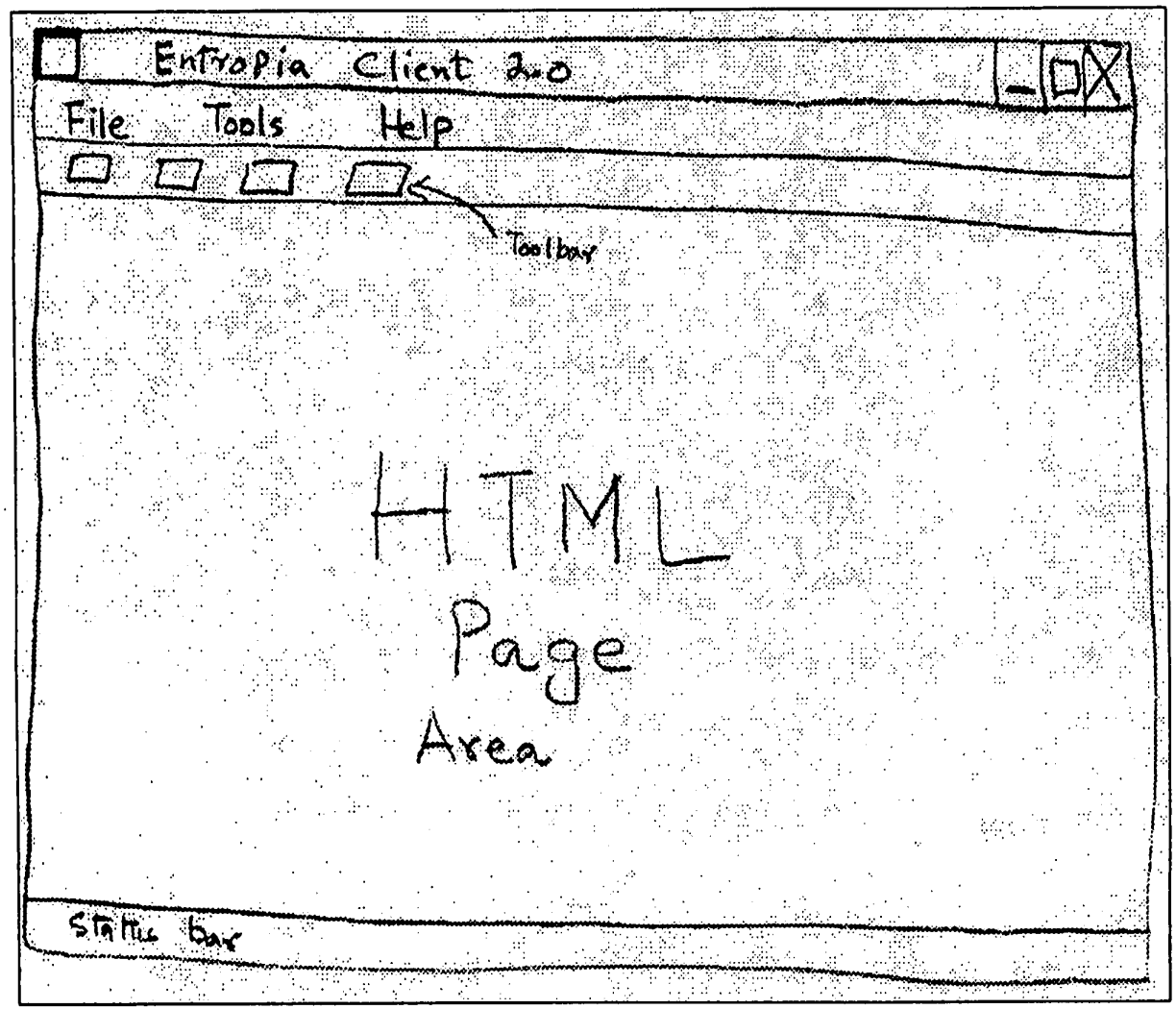
Entropia Client 2.0
File
Tools
Help
Toolbar
HTML
Page
Area
Status bar

FIG. 16

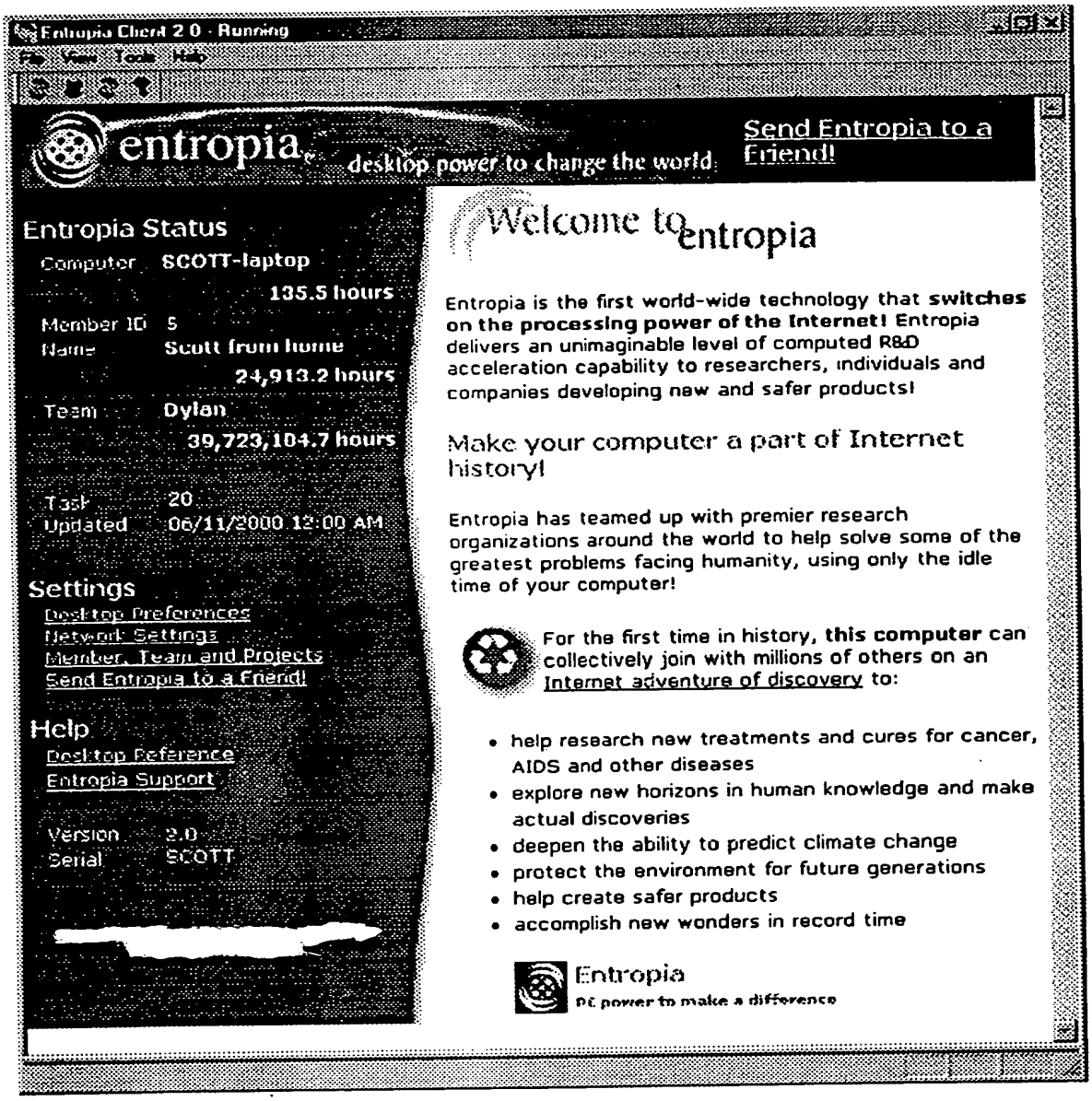
Entropia Client 2.0 - Running
File View Tools Help
entropia
desktop power to change the world
Send Entropia to a Friend!
Entropia Status
Computer SCOTT-laptop
135.5 hours
Member ID 5
Name Scott from home
24,913.2 hours
Team Dylan
39,723,104.7 hours
Task 20
Updated 06/11/2000 12:00 AM
Settings
Desktop Preferences
Network Settings
Member, Team and Projects
Send Entropia to a Friend!
Help
Desktop Reference
Entropia Support
Version 2.0
Serial SCOTT
Welcome to entropia
Entropia is the first world-wide technology that switches on the processing power of the Internet! Entropia delivers an unimaginable level of computed R&D acceleration capability to researchers, individuals and companies developing new and safer products!
Make your computer a part of Internet history!
Entropia has teamed up with premier research organizations around the world to help solve some of the greatest problems facing humanity, using only the idle time of your computer!
For the first time in history, this computer can collectively join with millions of others on an Internet adventure of discovery to:
help research new treatments and cures for cancer, AIDS and other diseases
explore new horizons in human knowledge and make actual discoveries
deepen the ability to predict climate change
protect the environment for future generations
help create safer products
accomplish new wonders in record time
Entropia
PC power to make a difference

FIG. 17

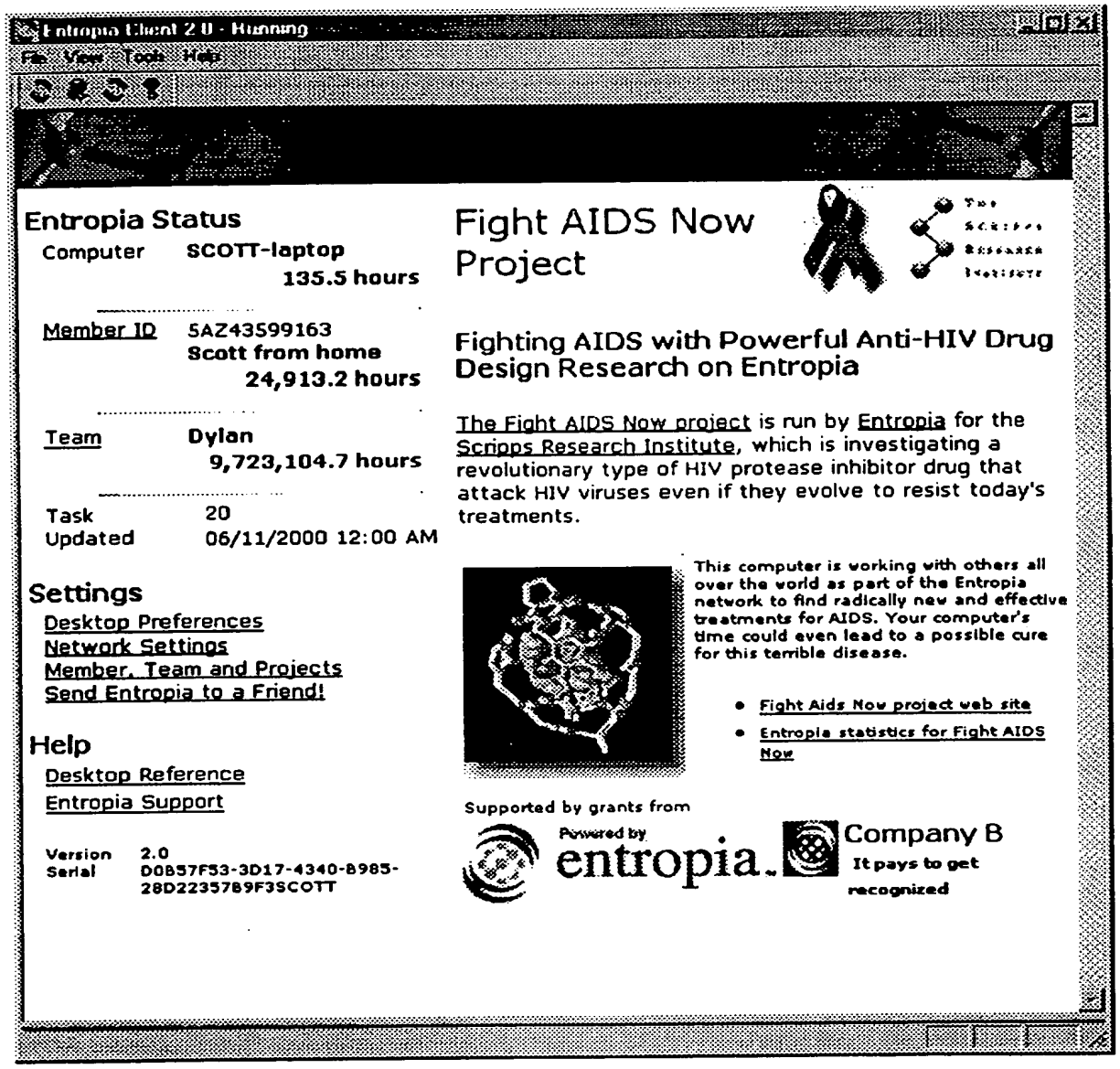
Entropia Client 2.0 - Running
File View Tools Help
Entropia Status
Computer SCOTT-laptop
135.5 hours
Member ID 5AZ43599163
Scott from home
24,913.2 hours
Team Dylan
9,723,104.7 hours
Task 20
Updated 06/11/2000 12:00 AM
Settings
Desktop Preferences
Network Settings
Member, Team and Projects
Send Entropia to a Friend!
Help
Desktop Reference
Entropia Support
Version 2.0
Serial D0B57F53-3D17-4340-8985-28D2235789F3SCOTT
Fight AIDS Now Project
Fighting AIDS with Powerful Anti-HIV Drug Design Research on Entropia
The Fight AIDS Now project is run by Entropia for the Scripps Research Institute, which is investigating a revolutionary type of HIV protease inhibitor drug that attack HIV viruses even if they evolve to resist today's treatments.
This computer is working with others all over the world as part of the Entropia network to find radically new and effective treatments for AIDS. Your computer's time could even lead to a possible cure for this terrible disease.
Fight Aids Now project web site
Entropia statistics for Fight AIDS Now
Supported by grants from
Powered by entropia.
Company B
It pays to get recognized

FIG. 18

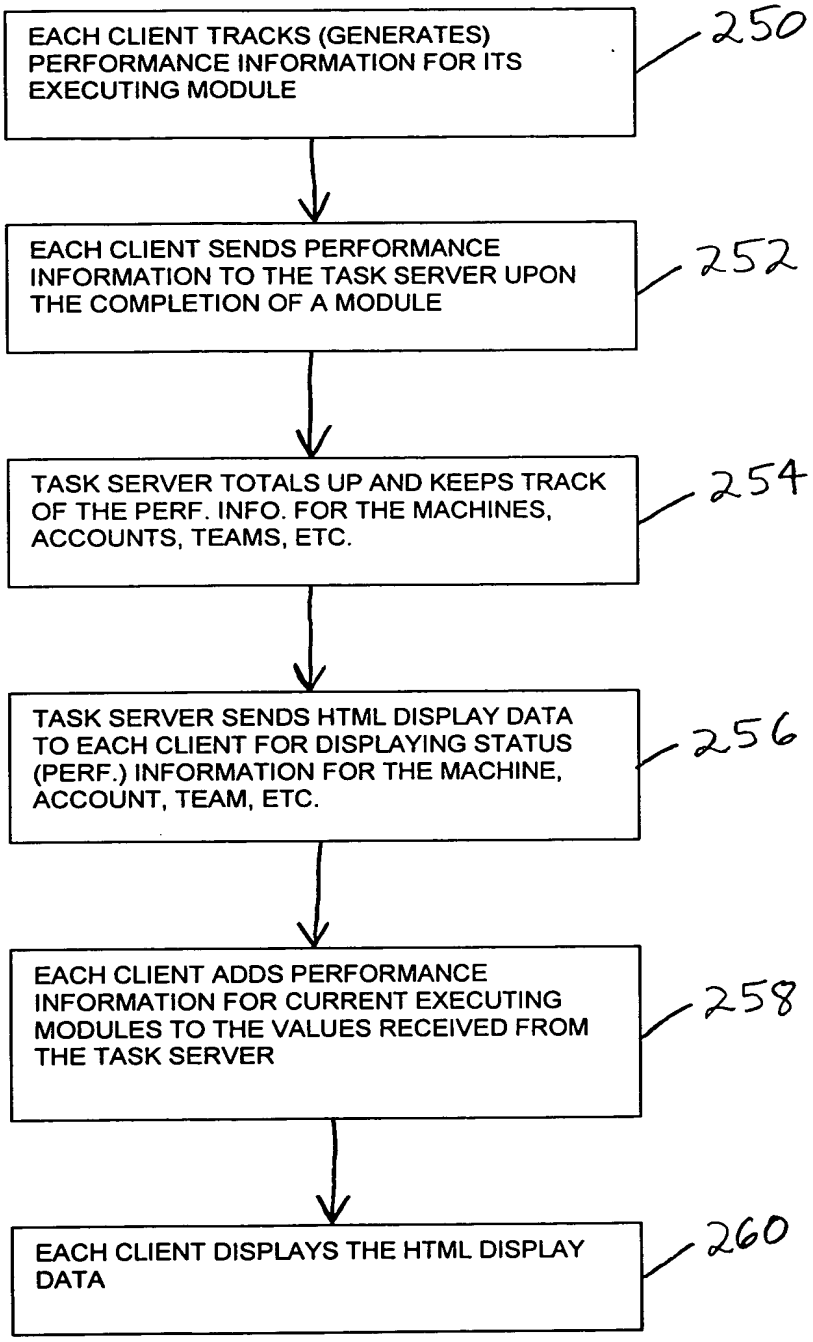
EACH CLIENT TRACKS (GENERATES) PERFORMANCE INFORMATION FOR ITS EXECUTING MODULE
250
EACH CLIENT SENDS PERFORMANCE INFORMATION TO THE TASK SERVER UPON THE COMPLETION OF A MODULE
252
TASK SERVER TOTALS UP AND KEEPS TRACK OF THE PERF. INFO. FOR THE MACHINES, ACCOUNTS, TEAMS, ETC.
254
TASK SERVER SENDS HTML DISPLAY DATA TO EACH CLIENT FOR DISPLAYING STATUS (PERF.) INFORMATION FOR THE MACHINE, ACCOUNT, TEAM, ETC.
256
EACH CLIENT ADDS PERFORMANCE INFORMATION FOR CURRENT EXECUTING MODULES TO THE VALUES RECEIVED FROM THE TASK SERVER
258
EACH CLIENT DISPLAYS THE HTML DISPLAY DATA
260

FIG. 19

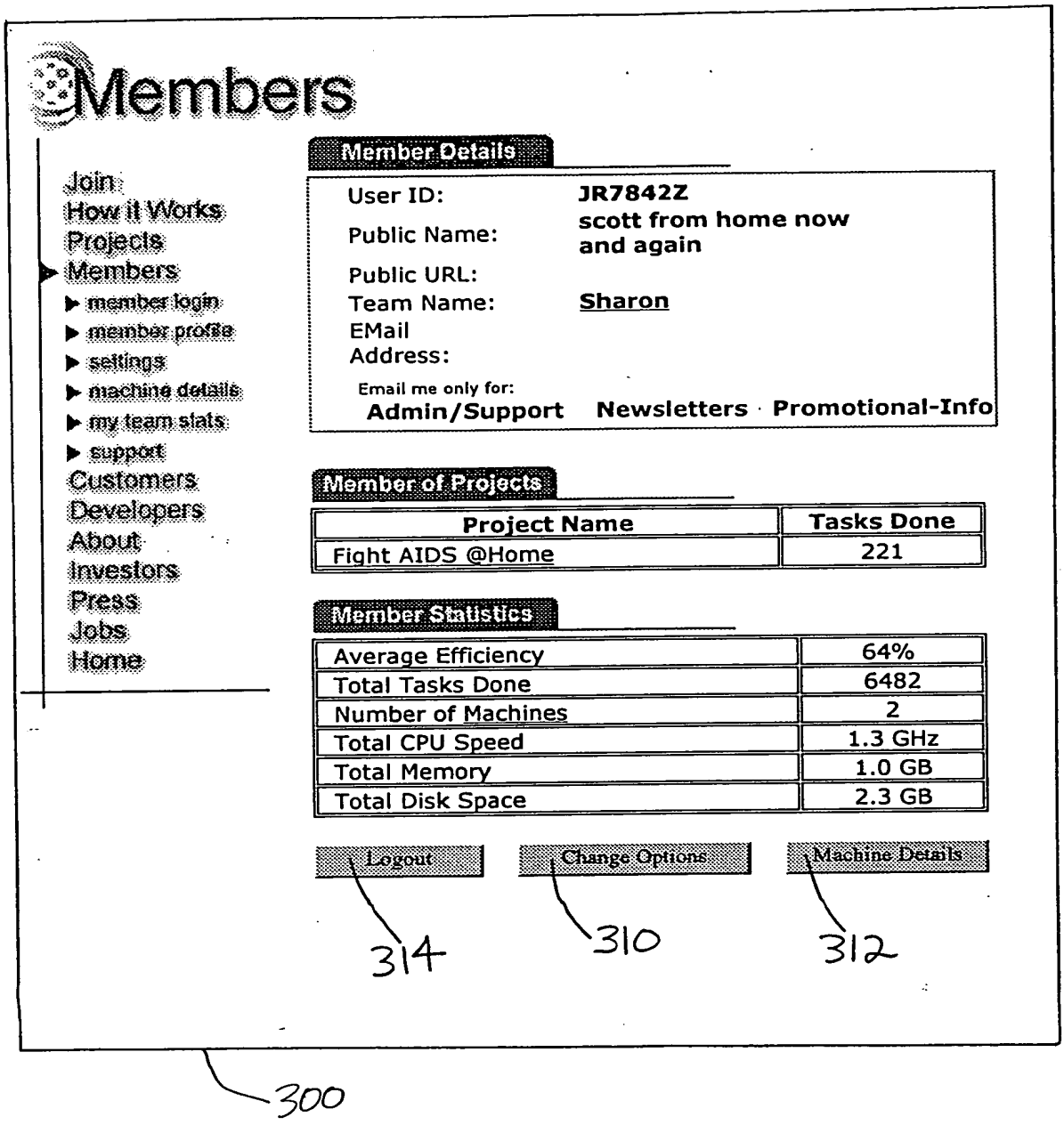
Members
Join
How it Works
Projects
Members
member login
member profile
settings
machine details
my team stats
support
Customers
Developers
About
Investors
Press
Jobs
Home
Member Details
User ID: JR7842Z
Public Name: scott from home now and again
Public URL:
Team Name: Sharon
EMail Address:
Email me only for:
Admin/Support Newsletters Promotional-Info
Member of Projects
Project Name | Tasks Done
Fight AIDS @Home | 221
Member Statistics
Average Efficiency | 64%
Total Tasks Done | 6482
Number of Machines | 2
Total CPU Speed | 1.3 GHz
Total Memory | 1.0 GB
Total Disk Space | 2.3 GB
Logout
Change Options
Machine Details
314
310
312
300

FIG. 20

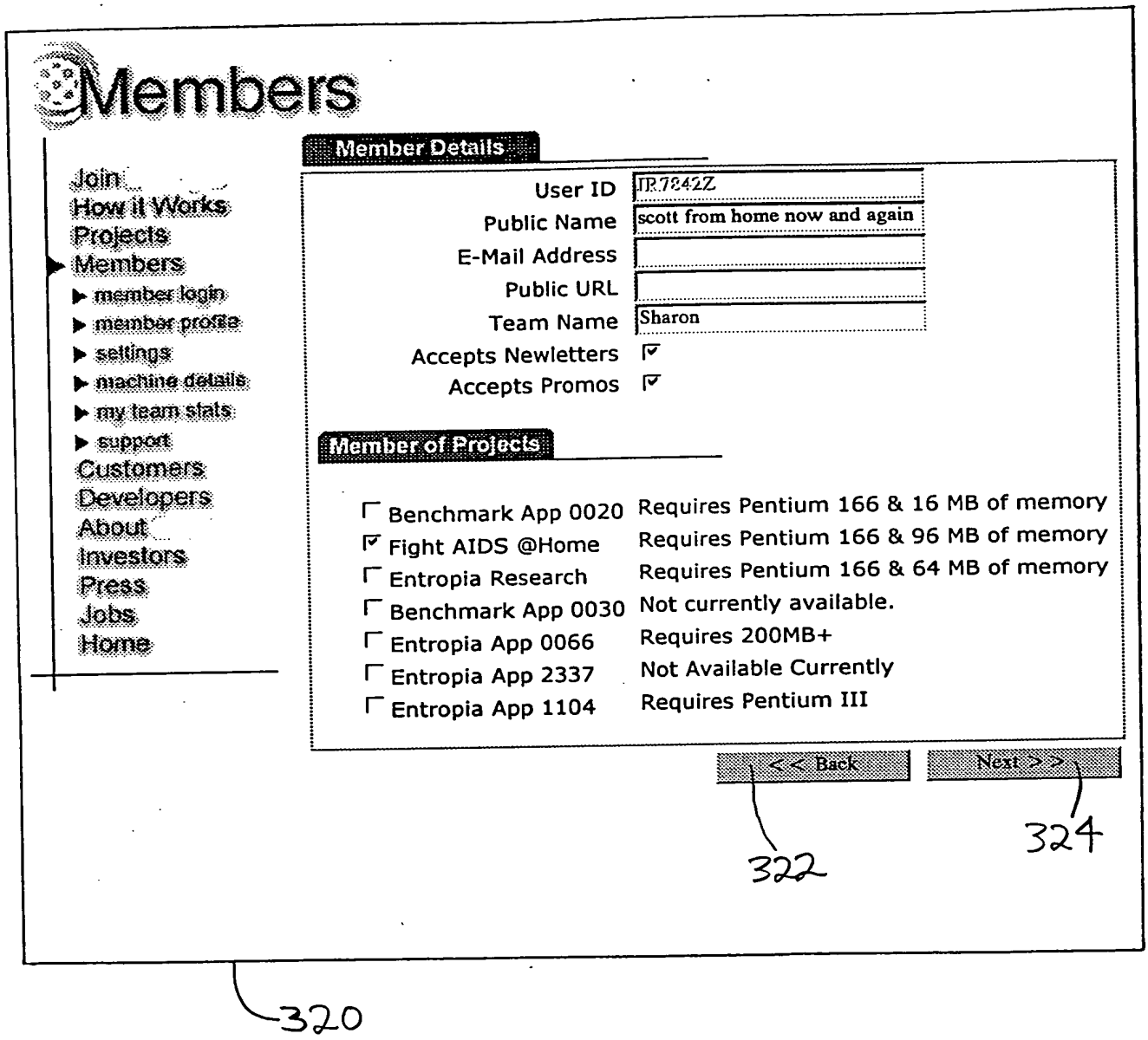
Members
Join
How it Works
Projects
Members
member login
member profile
settings
machine details
my team stats
support
Customers
Developers
About
Investors
Press
Jobs
Home
Member Details
User ID
IP7842Z
Public Name
scott from home now and again
E-Mail Address
Public URL
Team Name
Sharon
Accepts Newletters
Accepts Promos
Member of Projects
Benchmark App 0020
Requires Pentium 166 & 16 MB of memory
Fight AIDS @Home
Requires Pentium 166 & 96 MB of memory
Entropia Research
Requires Pentium 166 & 64 MB of memory
Benchmark App 0030
Not currently available.
Entropia App 0066
Requires 200MB+
Entropia App 2337
Not Available Currently
Entropia App 1104
Requires Pentium III
<< Back
Next >>
322
324
320

FIG. 21

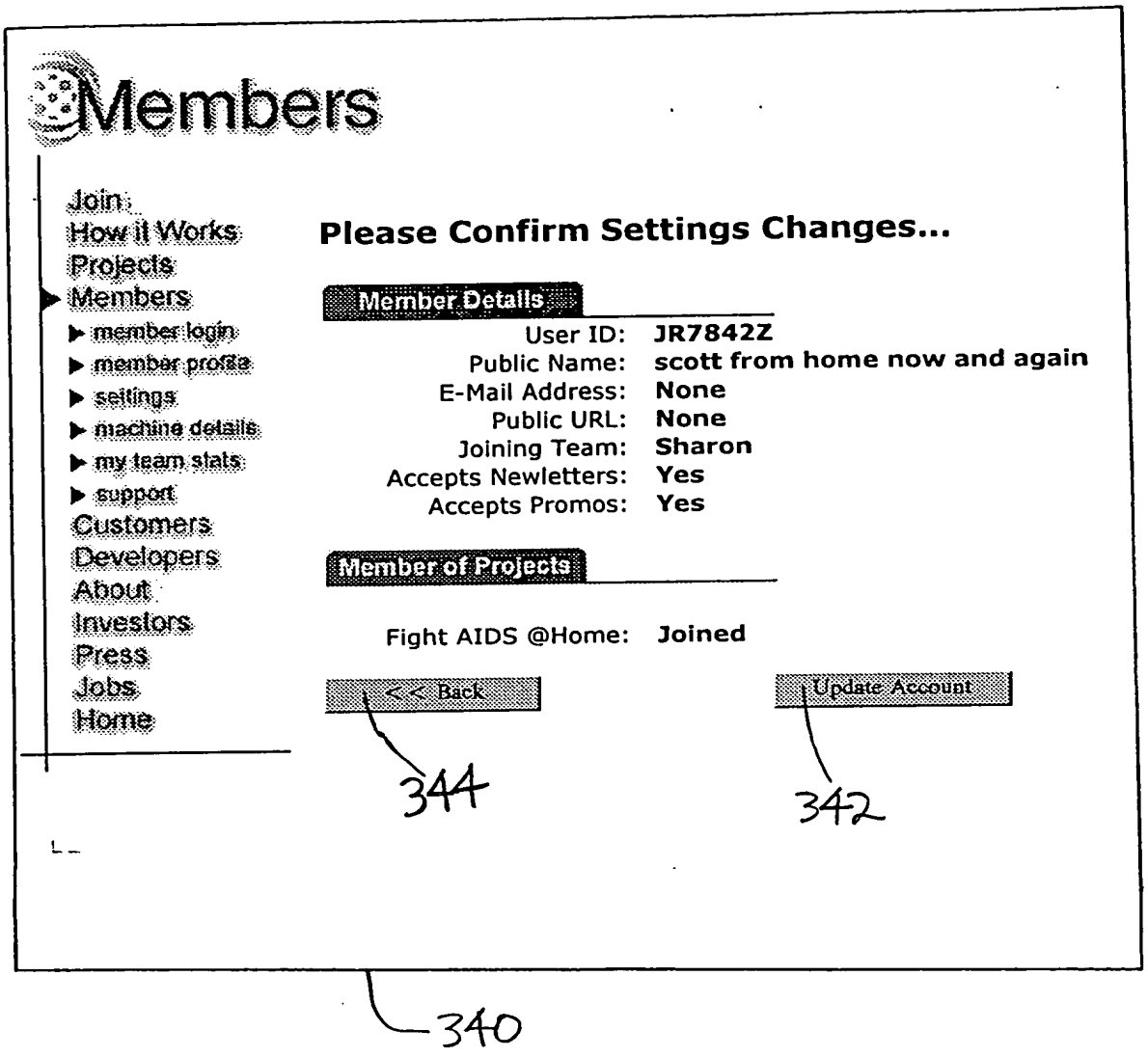
Members
Join
How it Works
Projects
Members
member login
member profile
settings
machine details
my team stats
support
Customers
Developers
About
Investors
Press
Jobs
Home
Please Confirm Settings Changes...
Member Details
User ID: JR7842Z
Public Name: scott from home now and again
E-Mail Address: None
Public URL: None
Joining Team: Sharon
Accepts Newletters: Yes
Accepts Promos: Yes
Member of Projects
Fight AIDS @Home: Joined
<< Back
Update Account
344
342
340

FIG. 22

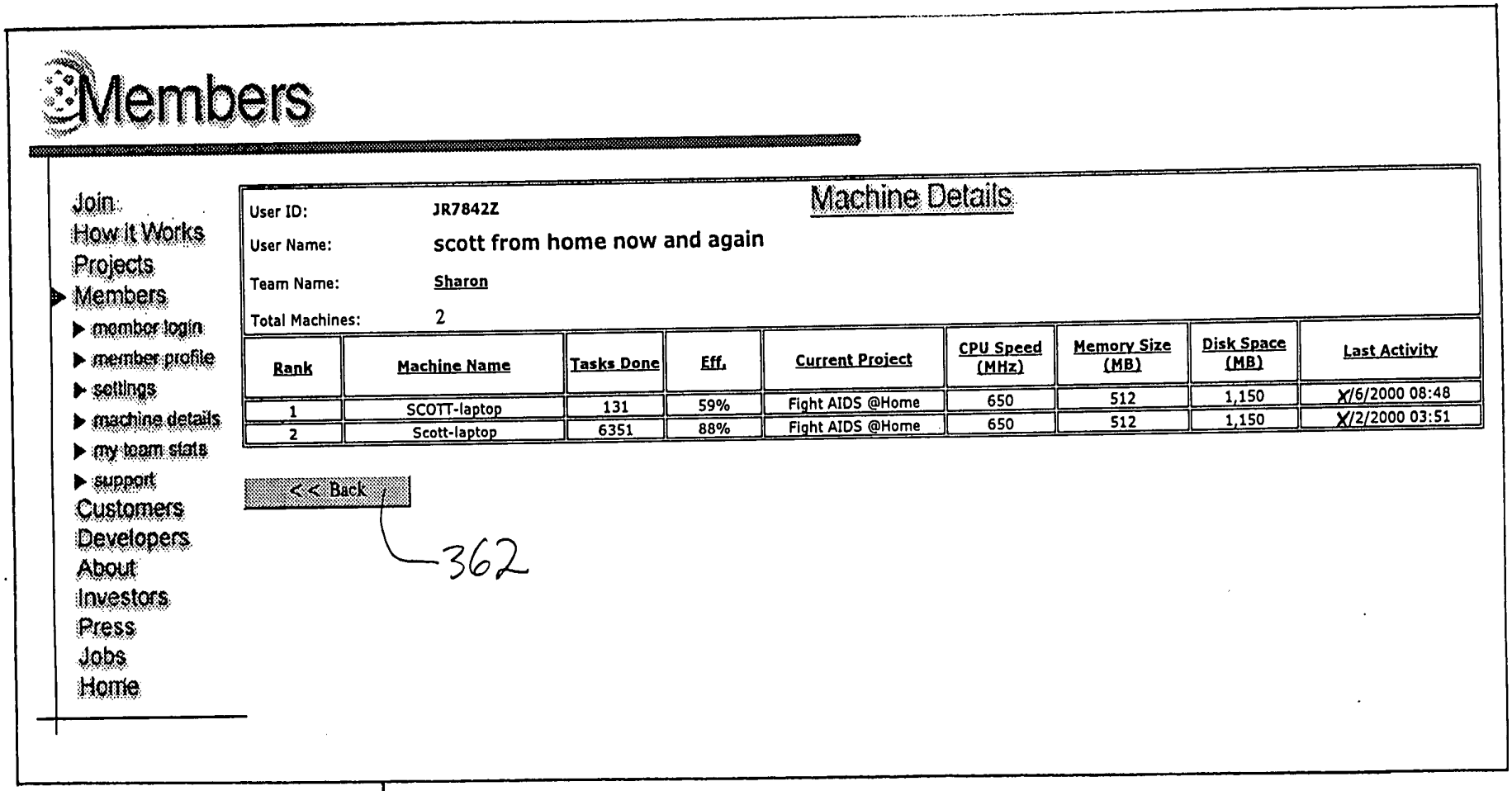
Members
Join
How it Works
Projects
Members
member login
member profile
settings
machine details
my team stats
support
Customers
Developers
About
Investors
Press
Jobs
Home
Machine Details
User ID: JR7842Z
User Name: scott from home now and again
Team Name: Sharon
Total Machines: 2
Rank | Machine Name | Tasks Done | Eff. | Current Project | CPU Speed (MHz) | Memory Size (MB) | Disk Space (MB) | Last Activity
1 | SCOTT-laptop | 131 | 59% | Fight AIDS @Home | 650 | 512 | 1,150 | X/6/2000 08:48
2 | Scott-laptop | 6351 | 88% | Fight AIDS @Home | 650 | 512 | 1,150 | X/2/2000 03:51
<< Back
362
360

FIG. 23

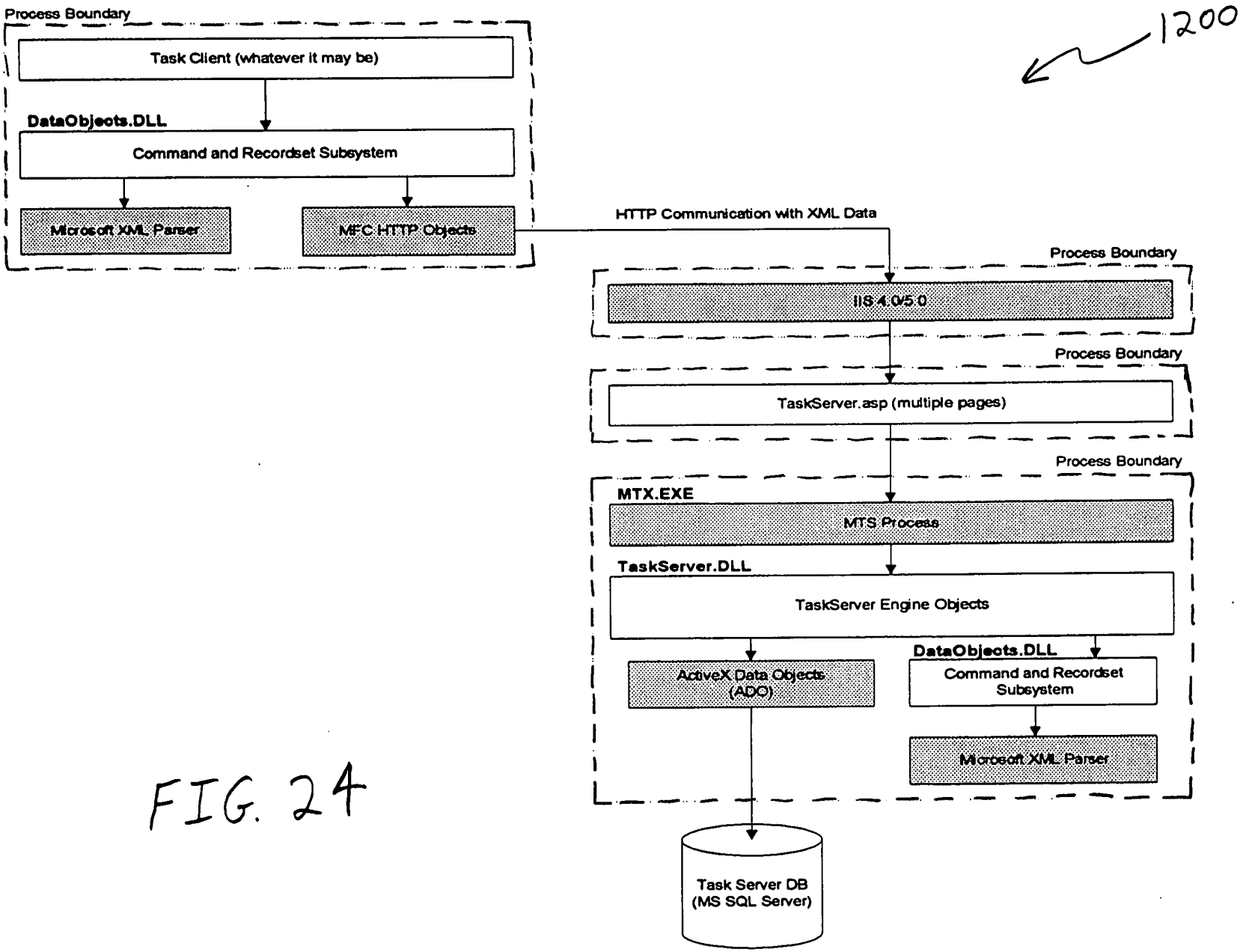
1200
Process Boundary
Task Client (whatever it may be)
DataObjects.DLL
Command and Recordset Subsystem
Microsoft XML Parser
MFC HTTP Objects
HTTP Communication with XML Data
Process Boundary
IIS 4.0/5.0
Process Boundary
TaskServer.asp (multiple pages)
Process Boundary
MTX.EXE
MTS Process
TaskServer.DLL
TaskServer Engine Objects
ActiveX Data Objects (ADO)
DataObjects.DLL
Command and Recordset Subsystem
Microsoft XML Parser
Task Server DB (MS SQL Server)

FIG. 24

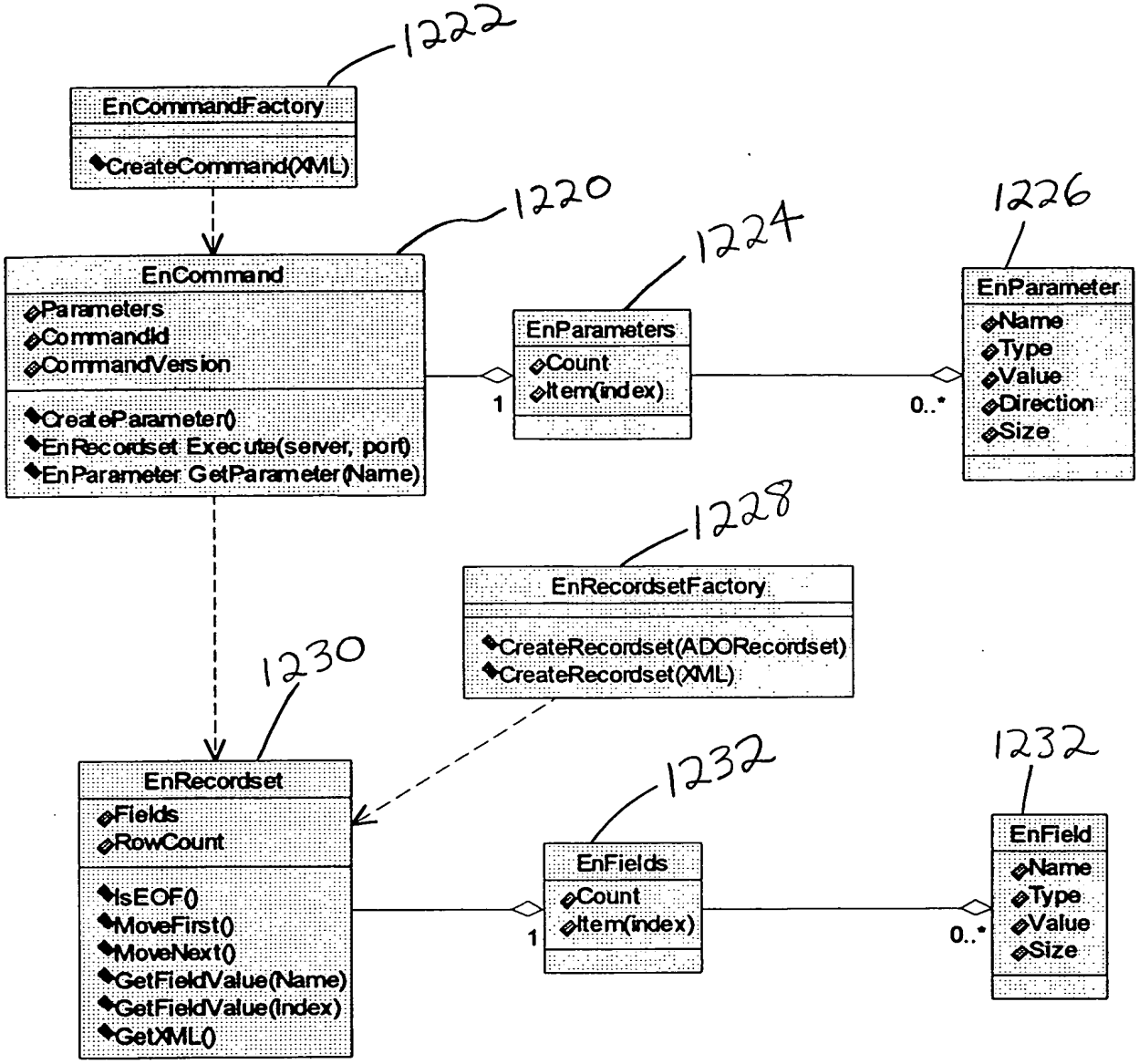
1222
EnCommandFactory
CreateCommand(XML)
1220
EnCommand
Parameters
CommandId
CommandVersion
CreateParameter()
EnRecordset Execute(server, port)
EnParameter GetParameter(Name)
1224
EnParameters
Count
Item(index)
1
0..*
1226
EnParameter
Name
Type
Value
Direction
Size
1228
EnRecordsetFactory
CreateRecordset(ADORecordset)
CreateRecordset(XML)
1230
EnRecordset
Fields
RowCount
IsEOF()
MoveFirst()
MoveNext()
GetFieldValue(Name)
GetFieldValue(Index)
GetXML()
1232
EnFields
Count
Item(index)
1
0..*
1232
EnField
Name
Type
Value
Size

FIG. 25

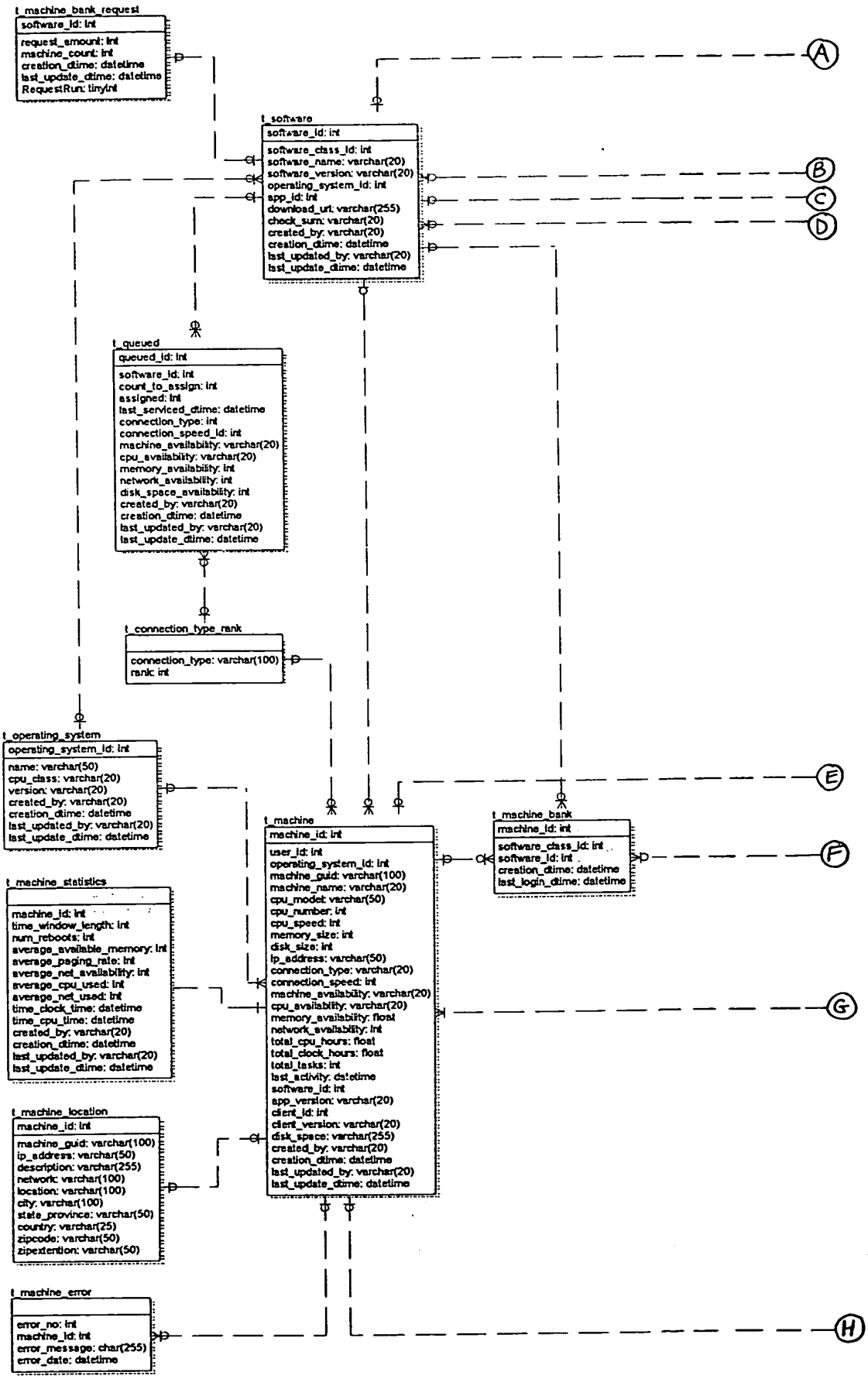
t_machine_bank_request
software_id: int
request_amount: int
machine_count: int
creation_dtime: datetime
last_update_dtime: datetime
RequestRun: tinyint
t_software
software_id: int
software_class_id: int
software_name: varchar(20)
software_version: varchar(20)
operating_system_id: int
app_id: int
download_url: varchar(255)
check_sum: varchar(20)
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_queued
queued_id: int
software_id: int
count_to_assign: int
assigned: int
last_serviced_dtime: datetime
connection_type: int
connection_speed_id: int
machine_availability: varchar(20)
cpu_availability: varchar(20)
memory_availability: int
network_availability: int
disk_space_availability: int
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_connection_type_rank
connection_type: varchar(100)
rank: int
t_operating_system
operating_system_id: int
name: varchar(50)
cpu_class: varchar(20)
version: varchar(20)
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_machine
machine_id: int
user_id: int
operating_system_id: int
machine_guid: varchar(100)
machine_name: varchar(20)
cpu_model: varchar(50)
cpu_number: int
cpu_speed: int
memory_size: int
disk_size: int
ip_address: varchar(50)
connection_type: varchar(20)
connection_speed: int
machine_availability: varchar(20)
cpu_availability: varchar(20)
memory_availability: float
network_availability: int
total_cpu_hours: float
total_clock_hours: float
total_tasks: int
last_activity: datetime
software_id: int
app_version: varchar(20)
client_id: int
client_version: varchar(20)
disk_space: varchar(255)
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_machine_bank
machine_id: int
software_class_id: int
software_id: int
creation_dtime: datetime
last_login_dtime: datetime
t_machine_statistics
machine_id: int
time_window_length: int
num_reboots: int
average_available_memory: int
average_paging_rate: int
average_net_availability: int
average_cpu_used: int
average_net_used: int
time_clock_time: datetime
time_cpu_time: datetime
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_machine_location
machine_id: int
machine_guid: varchar(100)
ip_address: varchar(50)
description: varchar(255)
network: varchar(100)
location: varchar(100)
city: varchar(100)
state_province: varchar(50)
country: varchar(25)
zipcode: varchar(50)
zipextention: varchar(50)
t_machine_error
error_no: int
machine_id: int
error_message: char(255)
error_date: datetime
A
B
C
D
E
F
G
H

FIG. 26A

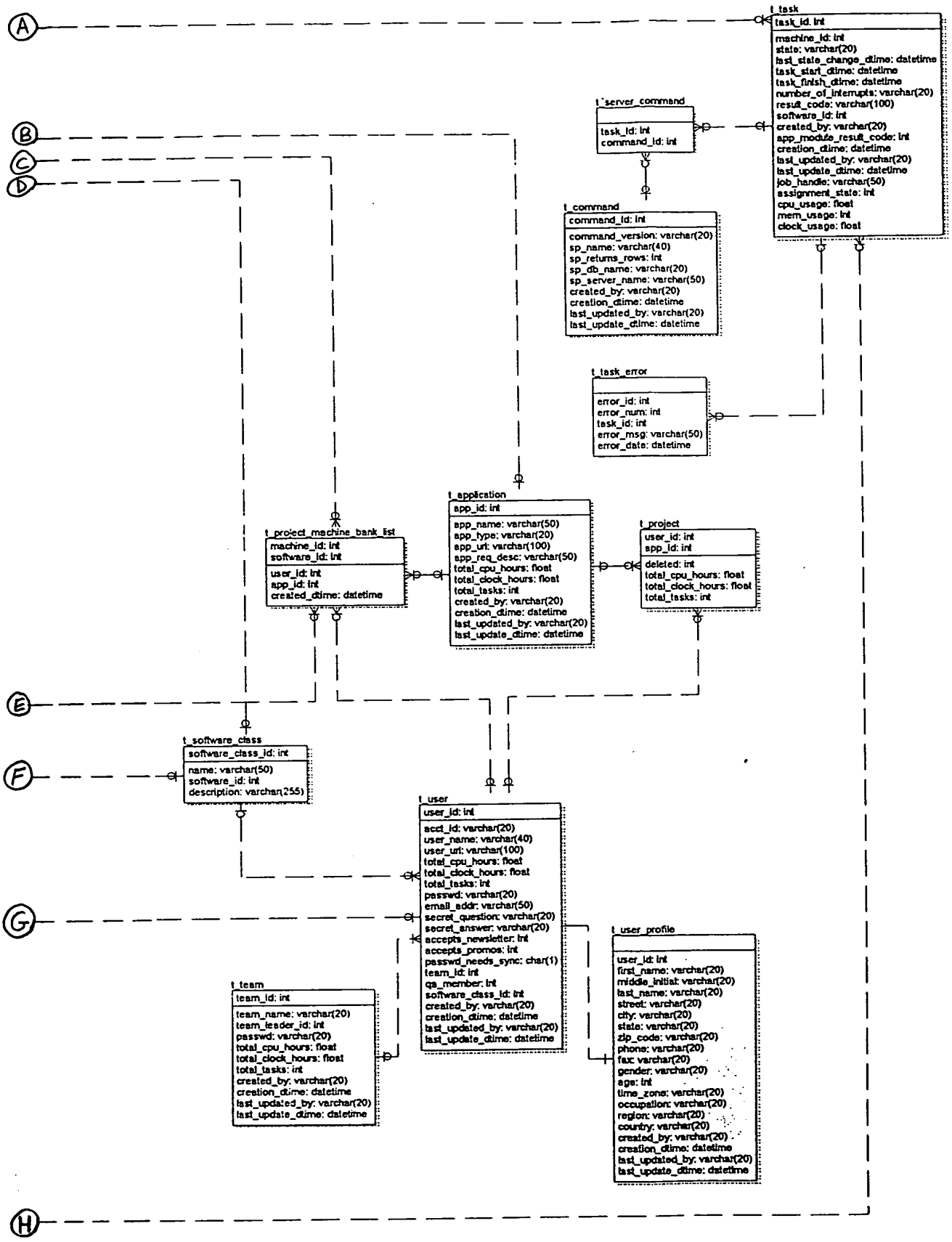
A
B
C
D
E
F
G
H
t_task
task_id: int
machine_id: int
state: varchar(20)
last_state_change_dtime: datetime
task_start_dtime: datetime
task_finish_dtime: datetime
number_of_interrupts: varchar(20)
result_code: varchar(100)
software_id: int
created_by: varchar(20)
app_module_result_code: int
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
job_handle: varchar(50)
assignment_state: int
cpu_usage: float
mem_usage: int
clock_usage: float
t_server_command
task_id: int
command_id: int
t_command
command_id: int
command_version: varchar(20)
sp_name: varchar(40)
sp_returns_rows: int
sp_db_name: varchar(20)
sp_server_name: varchar(50)
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_task_error
error_id: int
error_num: int
task_id: int
error_msg: varchar(50)
error_date: datetime
t_application
app_id: int
app_name: varchar(50)
app_type: varchar(20)
app_url: varchar(100)
app_req_desc: varchar(50)
total_cpu_hours: float
total_clock_hours: float
total_tasks: int
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_project_machine_bank_list
machine_id: int
software_id: int
user_id: int
app_id: int
created_dtime: datetime
t_project
user_id: int
app_id: int
deleted: int
total_cpu_hours: float
total_clock_hours: float
total_tasks: int
t_software_class
software_class_id: int
name: varchar(50)
software_id: int
description: varchar(255)
t_user
user_id: int
acct_id: varchar(20)
user_name: varchar(40)
user_url: varchar(100)
total_cpu_hours: float
total_clock_hours: float
total_tasks: int
passwd: varchar(20)
email_addr: varchar(50)
secret_question: varchar(20)
secret_answer: varchar(20)
accepts_newsletter: int
accepts_promos: int
passwd_needs_sync: char(1)
team_id: int
qa_member: int
software_class_id: int
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_user_profile
user_id: int
first_name: varchar(20)
middle_initial: varchar(20)
last_name: varchar(20)
street: varchar(20)
city: varchar(20)
state: varchar(20)
zip_code: varchar(20)
phone: varchar(20)
fax: varchar(20)
gender: varchar(20)
age: int
time_zone: varchar(20)
occupation: varchar(20)
region: varchar(20)
country: varchar(20)
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime
t_team
team_id: int
team_name: varchar(20)
team_leader_id: int
passwd: varchar(20)
total_cpu_hours: float
total_clock_hours: float
total_tasks: int
created_by: varchar(20)
creation_dtime: datetime
last_updated_by: varchar(20)
last_update_dtime: datetime

FIG. 26B